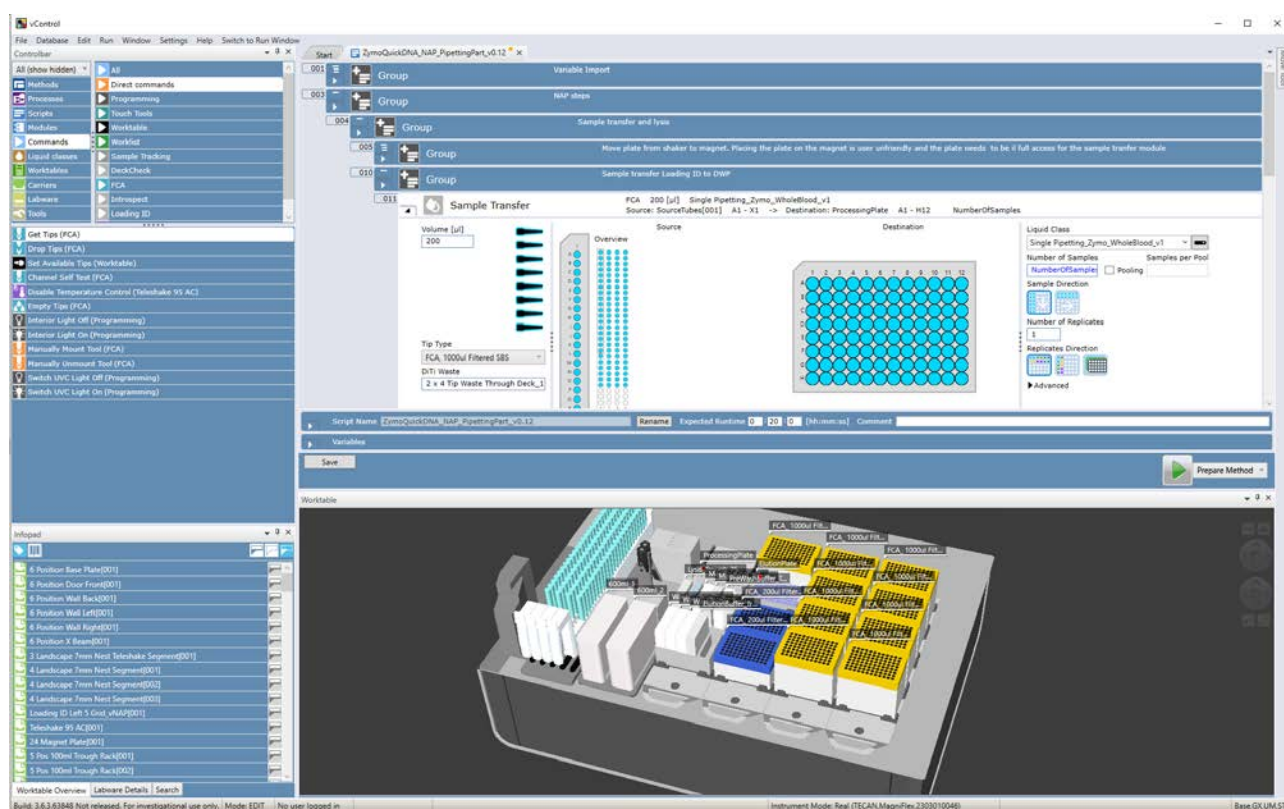


Application Software Manual

vControl



Title:	vControl Application Software Manual		Part number:	30250582.02
ID:	402665, en, Version 1.2		Translated from:	n.a.
Version:	Revision:	Issue:	Document History:	
1	0	2024-02-15	First edition	
1	1	2024-09-09	Chapter 1.2, 1.5.1, 1.6 updated Chapter 2.7 removed	
1	2	2024-11-28	Chapter 4.2, 5.7.5, 7.6.10, 8.3.8, 8.6.3, 8.6.7, 9.1, 9.1.3, 11.11.4 and 15 newly added or modified	

Table of Contents

1 About This Manual	9
1.1 Scope of This Manual	9
1.2 Intended Use	9
1.3 Improper Use	9
1.4 Trademarks	9
1.5 Reference Documents	9
1.6 License Information	10
1.7 Camera Privacy Statement	10
1.8 Document Conventions	10
2 Safety.....	11
2.1 Safety Message Conventions	11
2.2 General Safety Information	13
2.3 Operating Company	14
2.4 Method and Process Validation	14
2.5 User Qualification	15
2.6 Safety Elements	16
2.7 Optical Radiation (UVC)	17
2.8 Decontamination Declaration	17
3 Description.....	18
3.1 Overview	18
3.2 Supported Instruments and Options	18
4 Installation	19
4.1 Computer Requirements	19
4.2 Installing Application Software	20
4.3 Activate License	29
5 Getting Started	32
5.1 Starting the Software	32

5.2	Teaching with the MoveTool	32
5.3	Starting vControl	33
5.4	Logging In	33
5.5	The Configuration Editor	34
5.6	TouchTools	46
5.7	Scripts	49
5.8	Modules.....	61
5.9	Moving the Arms (Task of Path Finder)	61
5.10	PathFinder and Robot Vectors.....	63
5.11	Lid Handling	67
5.12	Regrip.....	68
5.13	Simulation Mode	70
6	User Administration	74
6.1	Main Features	74
6.2	Login Settings	76
6.3	Password Settings	77
6.4	Notify Settings	78
6.5	User – Role permission.....	79
6.6	Role – Method permission	81
6.7	Roles and Access Rights	81
6.8	Electronic Signature	85
6.9	Deactivate User Management.....	86
6.10	User Management Configuration Options.....	86
7	Menus	92
7.1	File	92
7.2	Database.....	93
7.3	Edit.....	98
7.4	Run.....	99
7.5	Window	100
7.6	Settings	100

7.7	Help.....	103
7.8	Switch to Run Window Button.....	105
8	Editor.....	106
8.1	Worktable Editor.....	106
8.2	Carrier Editor.....	115
8.3	Labware Editor.....	128
8.4	Script Editor.....	140
8.5	Method Editor.....	143
8.6	Liquid Class Editor.....	147
9	Configure Your System	156
9.1	Teaching	157
9.2	Configuring Your Drivers.....	159
9.3	Other System	162
10	Validation Checklist	164
11	Commands.....	169
11.1	Direct Commands	169
11.2	FCA.....	176
11.3	FCA Gripper	216
11.4	RGA	220
11.5	Loading ID for vControl	225
11.6	DeckCheck.....	234
11.7	Magellan.....	241
11.8	Inheco Devices.....	243
11.9	Programming.....	247
11.10	Touch Tools	264
11.11	Worklist	282
11.12	Sample Tracking	296
11.13	Introspect	301
11.14	Worktable	302

12 Functions and Operators.....	306
12.1 Operators Supported by Expression Solver.....	306
12.2 Functions.....	308
12.3 Operands	316
12.4 Worktable Lookup Functions.....	316
 13 Data Handling	 323
13.1 Saving User Data	323
13.2 Backup	323
13.3 Restoring.....	324
13.4 Archiving Log and Audit Trail Data.....	324
 14 Error Handling and Recovery.....	 325
14.1 Errors Requiring User Interaction (FCA).....	325
14.2 Error without User Interaction (FCA).....	345
14.3 Error Requiring User Interaction (RGA)	347
14.4 Error Requiring User Interaction (Loading ID).....	349
 15 Method Recovery	 351
15.1 How to Setup and Run Method Recovery.....	351
15.2 Script Commands.....	353
 16 API	 361
16.1 Execution Channels	361
16.2 Functions.....	362
16.3 Examples	365
 17 Scheduler	 370
17.1 The Process Editor.....	370
17.2 Define Methods Including Processes.....	380
17.3 Running a Process.....	382
17.4 User Rights	382
17.5 Scheduler Configuration.....	382

17.6 The Labware Group Editor	384
17.7 Devices	384
17.8 Automated Tip Handling.....	385
17.9 Validating a Process	386
17.10 Optimizing a Process	387
17.11 Support for Cell-Based Assays	387
17.12 Not Supported	390
18 Utilities	391
18.1 Log Viewer	391
18.2 OLEG Log Viewer	392
19 Customer Support.....	393
19.1 Data Collection for Customer Support	393
19.2 Contacts	405
20 Abbreviations	408
20.1 Abbreviations Table	408
Glossary	410

1 About This Manual

This Application Software Manual describes the vControl software, provides information required for its proper installation, and instructs how to run the software and how to make use of all its features.

Before installing and running the vControl software, read this manual carefully—in particular section [“Safety” \[▶ 11\]](#).

1.1 Scope of This Manual

This manual applies to vControlApplication Software 1.0 or higher.

1.2 Intended Use

Please refer to manuals listed in [“Chapter 1.5.1” \[▶ 9\]](#).

1.3 Improper Use

Improper use may prejudice the Veya safety concept.

- The Veya must not be used with options or components that are not approved by Tecan.
- The Veya is not explosion-proof and should not be installed in locations where there is a hazard of explosion.
- The Veya should not be used in the absence of functional safety devices.

1.4 Trademarks

The product names, whether registered or unregistered trademarks, mentioned in this manual are reproduced solely for identification purposes and remain the exclusive property of their respective owners. For simplicity reasons, the trademark symbols such as ® and ™ are not repeated in the manual.

1.5 Reference Documents

This section provides a list of the documents that are needed or may be useful when using the Veya.

The Doc IDs listed below are root numbers. Therefore, they do not contain information about the language, document version, or the medium (data storage medium, hard copy, downloadable file, etc.) of the document.



On the basis of your order configuration, the Operating Manuals for optional equipment apply as well.

Check the scope of the corresponding document to ensure that you are in possession of the correct version.

1.5.1 Instrument Manuals

- Veya Operating Manual (RUO, Doc ID 402664)
- Veya DX Operating Manual (IVD, DOC 403062)

1.5.2 Software Manuals

- Tecan Sample Tracking Add-on Software Manual (Doc ID 393933)
- For Tecan digital solutions, such as Introspect, see <https://www.tecan.com/knowledge-portal/digital-solutions>

1.6 License Information

Use the **About vControl** command on the **Help** menu for detailed information about licenses used in vControl.

Product includes software components of third parties whose source code is located in the subfolder C:\Program Files (x86)\Tecan\Third Party Source Code. For license overview please press **F2**.

1.7 Camera Privacy Statement

The Veya system is equipped with a camera mounted on the inside front profile. The camera is focused on the deck and rear deck. Views downwards through the acrylic-glass side panels is possible.

- The user is responsible for advising people in the room that cameras are in operation.
- The user is responsible for ensuring that the personnel could not be identified from pictures taken, for example if the instrument is adjacent (side-on) to a desk space or if rear or side panel cut-outs are made or if an acrylic-glass panel is used to replace the rear wall.

1.8 Document Conventions

Cross-References

Cross-references appear as follows—e.g.:

Refer to section “[Safety](#)” [▶ 11]

- “Safety” refers to the corresponding section header
- The page number is given in square brackets

Prerequisites

Prerequisites appear as follows—e.g.:

✓ “General Safety Information” has been read.

Tips

Additional tips appear as follows—e.g.:



For safety conventions and symbols refer to chapter “[Safety](#)” [▶ 11].

Illustrations

The illustrations may show component versions which are not relevant to your Veya.




2 Safety

This chapter describes the safety concept of Veya, provides general rules of correct behavior, and warnings concerning hazards associated with the use of the Veya.

2.1 Safety Message Conventions

2.1.1 Signal Words

Tab. 1: Signal Words

Signal Word	Meaning
 DANGER	Indicates a hazardous situation that, if not avoided, will result in death or serious injury.
 WARNING	Indicates a hazardous situation that, if not avoided, could result in death or serious injury.
 CAUTION	Indicates a hazardous situation that, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a situation that is not hazard-related but, if not avoided, could result in damage to or malfunctioning of the equipment, or incorrect process results.

2.1.2 Safety Symbols



Crushing of hands



General warning



Laser beam



Optical radiation



Biohazard



No heavy load



Magnetic Field

2.2 General Safety Information

WARNING

Veya is designed and built in accordance with the present state-of-the-art technology and the recognized technical safety regulations. Nevertheless, risks to users, property and the environment can arise if the Veya is used without due care and attention.

The safety of all users and personnel depends on the strict observation of these safety instructions and awareness of the safety-related warnings provided in this manual.

- Please pay great attention to the following general safety information.
 - This manual must always be available to all persons performing the tasks described herein.
 - Always use the power cable supplied with the instrument.
-
- Legal regulations, such as local, state and federal laws concerning the use or application, as well as the handling, of dangerous materials in connection with the Veya must be strictly followed.
 - The operating company is responsible for defining instructions in accordance with company procedures and local legal requirements. The instructions provided by the operating company must be strictly observed.
 - Observe the correct environmental conditions for storage and operation.
 - Structural changes to the safety devices are forbidden.
 - Damaged safety devices must be replaced immediately as described in this manual.
 - The Veya must not be modified in any way without prior consultation and written approval of Tecan. Authorized modifications to the system may only be performed by an FSE certified for the repair and upgrading of the Veya. Tecan will reject any claim resulting from unauthorized modifications.
 - Fire hazard caused by the improper use of the Veya. The Veya should not be installed in locations where there is a hazard of explosion.
 - Fire hazard caused by flammable liquids or system liquid.
 - Avoid the formation and accumulation of flammable vapors.
 - Chemical, biological, and radioactive hazards can be associated with the substances used or the samples and reagents processed with the Veya (e.g., during loading and unloading). The same applies to waste disposal.
 - Always be aware of possible hazards associated with these substances.
 - Use appropriate protective clothing, safety goggles, respirators, and gloves.
 - The handling of substances and the disposal of waste may be subject to local, state, or federal law, or to regulations with regard to health, environment, or safety. Strictly observe the corresponding provisions.
 - Any contamination must be dealt with immediately as described in this manual.
 - The user is responsible for ensuring that the Veya is always operated under proper conditions, and that maintenance, service, and repair tasks are performed with care, on schedule, and only by authorized personnel.

- Risk of incorrect measuring results. After system care or maintenance has been performed, operation must only be resumed after the correct system operating conditions have been verified.
- Always use recommended consumables within expiration date and original spare parts for maintenance and repair to assure good system performance and reliability.
- Injury could result if skin comes in contact with system liquid on the instrument.
 - Always wear protective clothing according to GLP.
- Heavy load! Do not lift the instrument.
- Do not operate the system without deck segments.
- If carry-over is not tolerated, the use of disposable tips with filters is strongly recommended.
- Possible Crash. Do not place devices without Tecan model data on the deck.
- The Veya is supplied with a biohazard safety sign which should be applied by the user in the event of use of biohazardous substances. Apply the label on the front door in a position visible to the user and convenient for the application. Refer to section Product Safety Signs.
- The ethernet cable of the DeckCheck cameras will be installed by an FSE and must be installed on the Veya PC at all times (EMC). The Ethernet interface is not allowed to be connected to a network.
- For California residents only: This product can expose you to chemicals such as lead which is known to the State of California to cause cancer and birth defects or other reproductive harm. For more information go to www.P65Warnings.ca.gov/product.

2.3 Operating Company

The operating company must ensure that the Veya and in particular the safety features, function properly and that all the personnel in contact with the instrument are adequately trained.

Responsi- bilities

- Method and process validation.
- Defining the processes in compliance with the Standard Operating Procedures.
- Ensuring that installation and operational qualifications (IQ OQs) have been completed.
- Ensuring that all personnel in contact with the Veya are adequately trained.
- Ensuring the availability of appropriate protective clothing and equipment.
- Ensuring the maintenance and safe operation of the Veya.
- Requiring adherence to laboratory safety regulations and directives.

2.4 Method and Process Validation

While performing method and process validation, pay attention to the following:

- Check that pipetted volumes meet the precision and accuracy requirements of the process being automated.
- When using non-Tecan or custom labware and aspirating with tracking, ensure that the container definition is correct (i.e., the appropriate speed is used for tracking) to avoid air aspiration.

**Responsibility
of the Key
Operator**

- Validate the application with regard to correct pipetting volumes and tracking.
- Validate the application to prevent aspiration too close to Z-max of custom labware.
- If chemicals and labware are not removed, the impact of UVC light on chemicals and labware present on the deck has to be evaluated and the assay validated.
- Include a manual post-run check for correct pipetting volumes.
- Personnel must be informed regarding the camera privacy statement (refer to Camera Privacy Statement).

2.5 User Qualification

The laboratory personnel must be fully qualified and trained to operate the Veya. The work described in this Application Software Manual must only be performed by authorized personnel with the qualifications prescribed below.

Laboratory personnel must:

- have suitable technical training,
- be familiar with the laboratory safety regulations and directives,
- be familiar with the instructions for the safety elements of the instrument,
- use protective clothing and equipment,
- be familiar with and adhere to good laboratory practices,
- and have read and understood the instructions in the Operating Manual.

Tecan recommends that the operator attends an operator training course. Please ask the Tecan Customer Service about available courses. Refer to section ["Customer Support" \[▶ 393\]](#).

2.5.1 Operator

Required Skills

The operator (lab technician) works for the operating company.

- No specific application or system knowledge
- Command of local languages
- Command of English is preferable

The operator has application software access rights allowing him to run methods and perform system care and will receive necessary training from the Key Operator.

2.5.2 Key Operator

Required Skills

The key operator (application specialist) supports the operating company or works for the same company.

- Extensive application knowledge
- Limited system knowledge
- Command of local languages
- Command of English
- In-depth knowledge of the corresponding software manual

**Responsi-
bilities**

- Instructing the operator
- Writing, running and validating methods

- Helping the operator to solve problems with the instrument

2.6 Safety Elements

CAUTION

Moving parts

The protection and safety elements installed on the Veya must not be removed, disabled or overridden during operation.

- If any devices are removed (e.g., for maintenance work), all protection and safety devices must be reinstalled, re-enabled and checked before resuming operations.

Safety panels and safety door locks are integral parts of the Veya.

2.6.1 Safety Panels

Veya is protected with safety panels:

The **front safety panel** can be opened and is fitted with door locks that cannot be opened during operation. During teaching the door locks are open in order to grant access to the worktable. A feature called Modulated speed ensures safety to the User is guaranteed. .

The drawer is also equipped with door locks, which can only be opened during a system pause or when the system is not in operation.

The **top and side safety panels** are fixed.

2.6.1.1 Front Safety Panels

The front safety panel prevents direct access to the robotic arms and to the elements on the instrument deck during operation. This is for the benefit of personal safety and improves method security. In addition, the front safety panel protects the user against spilling sample or reagent.

Full Front Safety Panel

The full front safety panel has the following features:

- No access to moving parts (moving parts, mechanical hazards)
- Protection of the samples against outside influence (method safety)
- Protection against spilling sample or reagent
- Protection against optical radiation (UVC)



With full front safety panels, only batch-wise loading is possible.

2.6.1.2 Safety Panels for Optional Devices

If an optional device is added to, or removed from, the side of the Veya, an appropriate side safety panel must be installed. Please consult the "[Customer Support](#)" [[▶ 393](#)].

2.6.2 Safety Panel Door Locks/Modulated Speed

The Veya safety concept assumes that the front safety panel is always closed when the instrument is running.

Active Stop

CAUTION! Unauthorized modifications to door locks are prohibited.



The following devices might not be interrupted by an active stop: Incubator, Magellan and Shaker. Interruption of other devices will depend on the device driver.

2.6.3 Drawer Door Locks

There are additional locks on the drawer accessing the disposable tip waste in order to protect the user from indirect access to the moving instrument.

2.7 Optical Radiation (UVC)

The Veya can be equipped with an optional UVC light option.

Exposure to UVC light radiation must be avoided as it can lead to injury. Door lock prevent access to the instrument while UVC light is running. A system pause, which opens the doorlocks will cause the UVC light to switch off automatically..

UVC light can be used in decontamination procedures. The suitability and effectiveness of using UVC for individual processes must be validated by the user.

2.8 Decontamination Declaration

In addition to regular system care, and in accordance with standard laboratory regulations, the Veya and its parts and accessories must be thoroughly decontaminated in the following circumstances:

- Before any maintenance or service work is performed on the Veya and, in particular, before an FSE intervention on the Veya
- In the event of accidents (e.g., crash, spillage, etc.)
- Before returning the Veya or its parts or accessories, to Tecan (e.g., for repair)
- Prior to storage
- Prior to disposal
- In general, before moving the Veya or its parts from its location

The owner of the instrument has full responsibility for the effective decontamination of all the equipment.

Before any intervention on the Veya by an FSE, and before returning the Veya or its parts or accessories to Tecan, the owner of the instrument must complete and sign the Decontamination Declaration form, confirming that the decontamination has been performed in accordance with good laboratory practice guidelines. Contact your local service organization to obtain this form and refer to section Decontamination.



Tecan reserves the right to refuse to deal with any Veya or its parts or accessories that is not accompanied by the Decontamination Declaration form.

3 Description

This chapter provides a brief overview of the vControl software and defines its scope of application. In addition, it describes the features and specifies the computer requirements for running the software.

vControl allows the user to interface, control and monitor the instrument.

3.1 Overview

vControl provides a graphical user interface for controlling Tecan's Veya series of pipetting instruments.

The vControl software allows the user to implement protocols (scripts) and to quickly visualize how a laboratory process or experiment can be implemented. Pipetting is modeled by individual script lines. Commands are provided for a wide range of Tecan hardware options and third-party devices and the scripting system is designed for maximum flexibility when developing complex custom methods.

3.2 Supported Instruments and Options

vControl is intended for the following instruments:

- Veya

vControl in this configuration supports the following optional devices, instrument options and systems:

- Flexible Channel Arm (FCA) with 4 or 8 channels and disposable tips, also with FCA gripper fingers
- Robotic Gripper Arm (RGA), also with eccentric gripper fingers
- Loading ID barcode scanner
- Inheco MTC devices
- Inheco ODTC
- Inheco Incubator (with or without shaking functionality)
- Mettler Toledo WXS weighing modules
- Tecan Magellan data reduction and analysis software for Tecan microplate readers
- Non-embedded PC (not purchased from Tecan)
- OneView (front screen)

4 Installation

This chapter describes the installation of the vControl software. Please follow the instructions carefully.



vControl must be installed on target computers that are or will be connected to the instrument.

To install vControl on the target computer, you need local administrator access rights for the Windows operating system.

4.1 Computer Requirements

To run vControl, your computer must meet the following requirements:

Component	Minimum requirements vControl
Operating System	Windows 10 Enterprise LTSC 2019, or Windows 10 Enterprise LTSC 2021
Processor*	Intel® Core™ i7 10th generation (Comet Lake) with UHD 630, or higher**
RAM	32 GB
Hard Drive	Solid-state drive (SSD) with 512 GB free space with high transfer speed (6 GB/s)
USB Ports	1 USB 2.0. Additional ports may be needed for additional devices
Network Interfaces	Two GigE Network cards with Intel® chipset for: <ul style="list-style-type: none"> • External network • Local network to any 3rd party devices and DeckCheck Optional: internet connectivity for remote support and Introspect.
Resolution of Monitor(s)	1920 x 1080

* It is recommended to use an Intel® Core™ i9 10th generation (Comet Lake) or higher for vControl Scheduler applications.

** Do not use a processor without integrated graphics for a DeckCheck configuration. Do not use a separate graphics card for a DeckCheck configuration.

NOTICE

Risk of the script or process aborting.

Risk of losing connection to the PC After a certain time of inactivity from the user, the PC goes into a sleep mode and the connection to the instrument may be lost. Adjust the PC settings to avoid from going into the sleep mode.

- The PC must be configured as to prevent it from going into **Sleep Mode**.
- The USB power management in Windows must be switched off to avoid losing the connection to the pipetting instrument.

Connect the pipetting instrument and other hardware options to the USB 2.0 port of the PC.



The use of an external USB hub should be avoided.

A passive external USB hub may generate data transfer interruptions and loss of data.

- If an external USB hub is absolutely necessary, use an active USB hub (i.e., with its own power supply).
- Please note that some port replicators for laptops contain a USB hub.

4.2 Installing Application Software

With the Tecan vControl Preinstaller, you install the necessary components to use vControl on your computer. Therefore, this step is mandatory and cannot be left out. After this step usually a system restart is required. The steps after the main installer are optional.



The Magellan driver requires the Tecan Magellan software, which must be installed separately.

Before installing vControl on your PC, make sure it meets the PC requirements.



Fig. 1: vControl AutoPlay Application Wizard

The installation consists of following steps:

- ReadMe
- Pre-Installation
- Install vControl
- Install 3D Sim
- Install Sample Tracking

To install the program, proceed as follows:

1. Click **Step 0: The ReadMe**
The ReadMe document opens. It contains information about how to upgrade the vControl software, remaining issues and new features.
2. Run the autorun application (autorun.exe) as administrator.
3. Click **Step 1 PreInstallation.**
The Tecan vControl Preinstaller appears:
4. Acknowledge the license terms and conditions, then click **Install.**
5. In the **User Account Control** dialog box, click **Yes** to allow the Tecan vControl Preinstaller Setup to make changes to your computer.

6. After you reach this screen, click **Close**.
7. Click **Step 2: Install vControl**.



Fig. 2: The Tecan vControl Setup Wizard

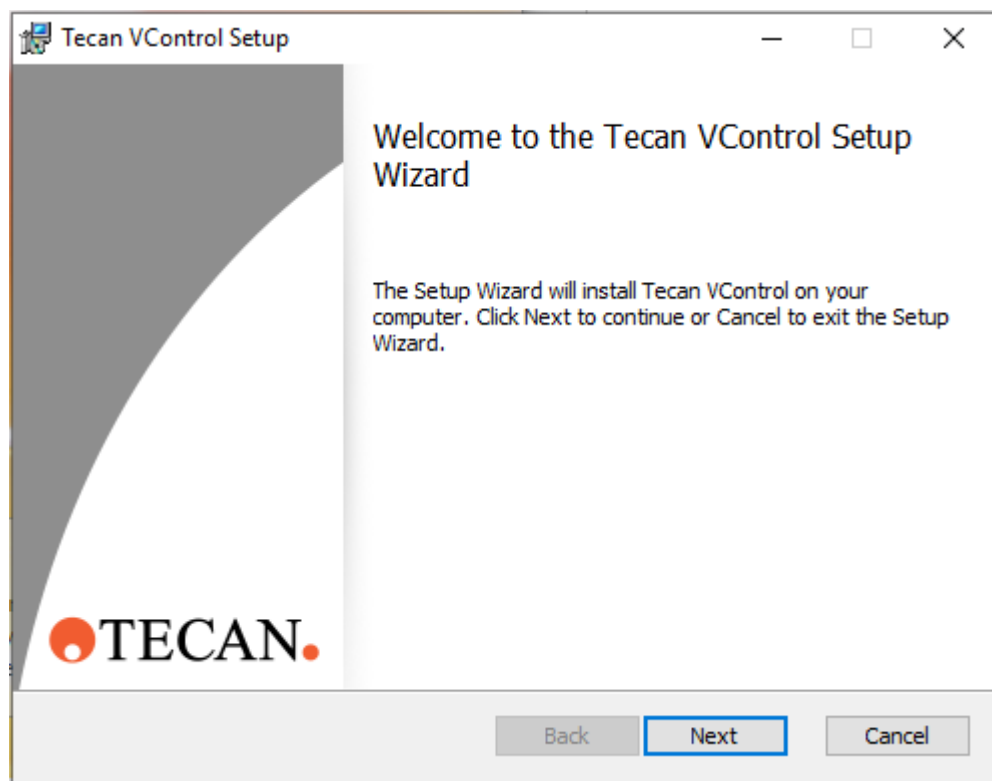
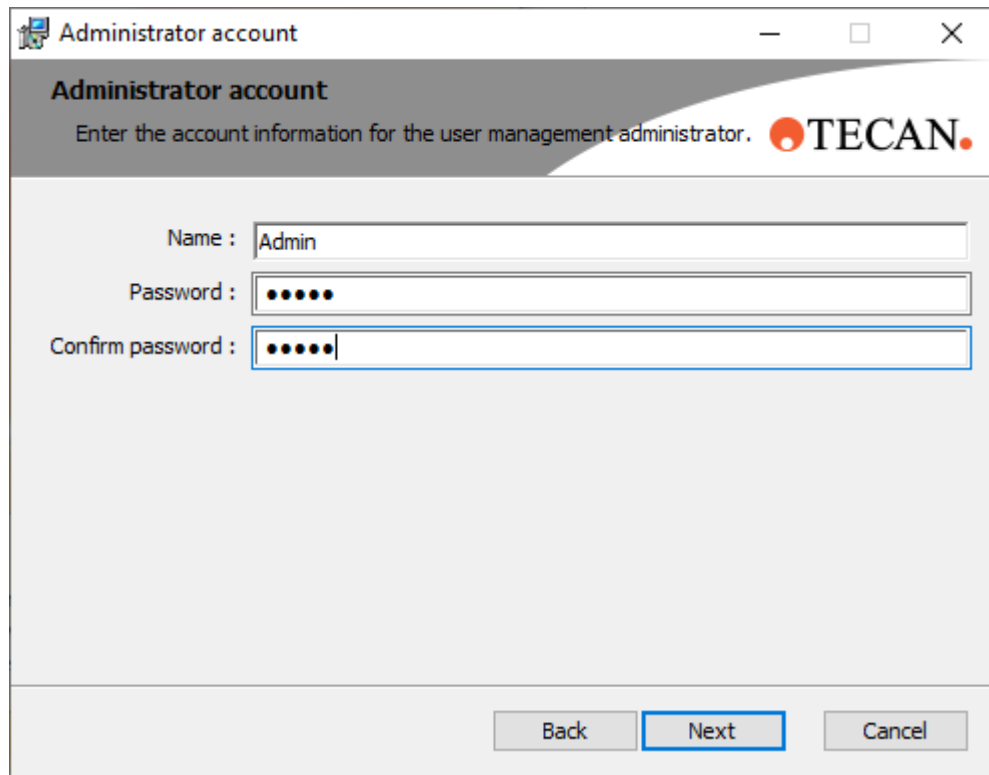


Fig. 3: Tecan vControl Main Installation Setup

8. Click **Next** in the Tecan vControl Setup screen.



The image shows a Windows-style dialog box titled "Administrator account". The title bar includes a small icon on the left and standard minimize, maximize, and close buttons on the right. The main content area has a header with the title "Administrator account" and a subtitle "Enter the account information for the user management administrator." followed by the TECAN logo. Below this, there are three input fields: "Name :" with the text "Admin" entered, "Password :" with five dots, and "Confirm password :" with five dots. At the bottom of the dialog, there are three buttons: "Back", "Next" (which is highlighted with a blue border), and "Cancel".

Fig. 4: Administrator account

9. Type in the name of the administrator and a password. The password must have at least 5 characters..
10. Confirm the password and click **Next**.

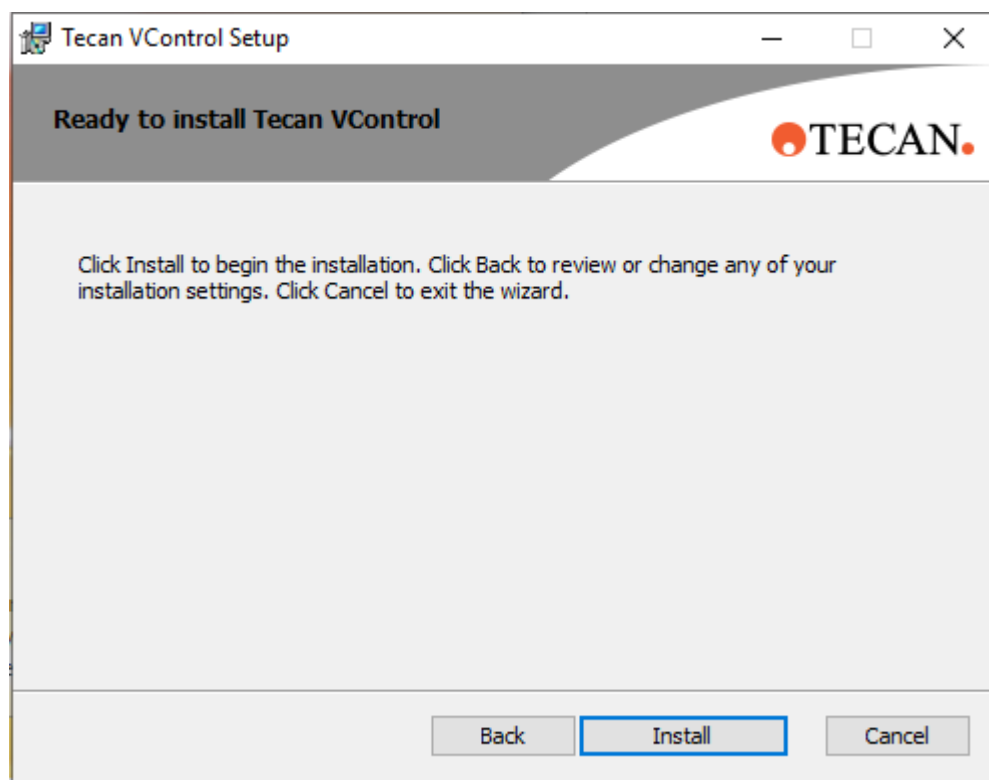


Fig. 5: Tecan vControl Setup: Ready to install

11. Click **Install**. The installation procedure may take some time.

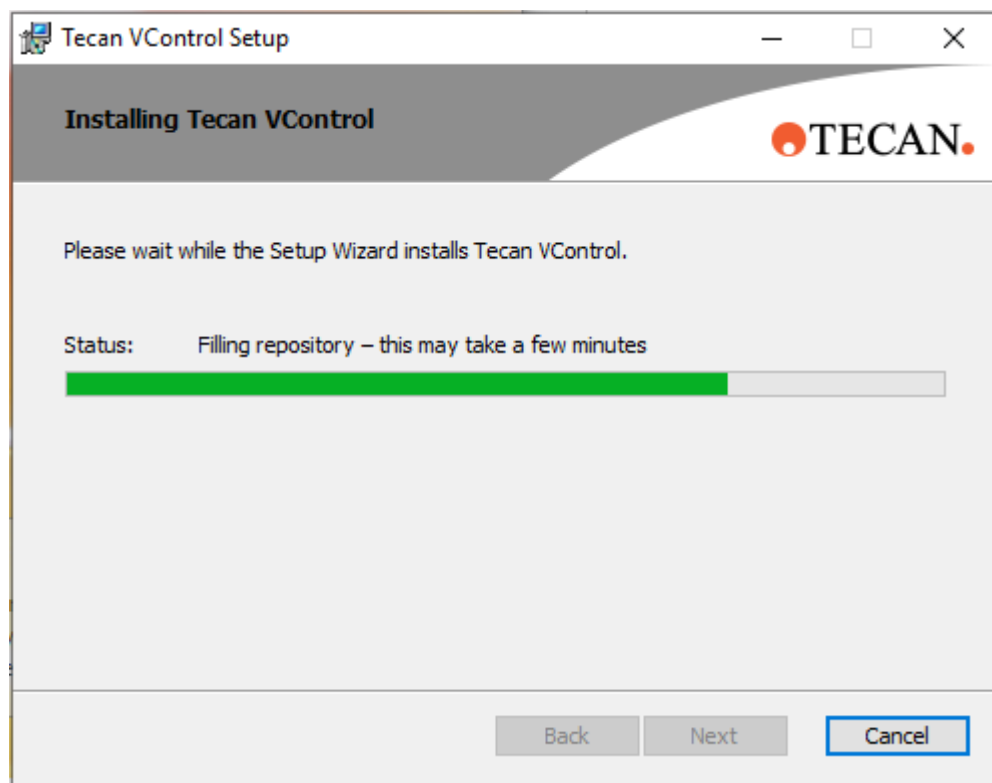


Fig. 6: nstallation vControl

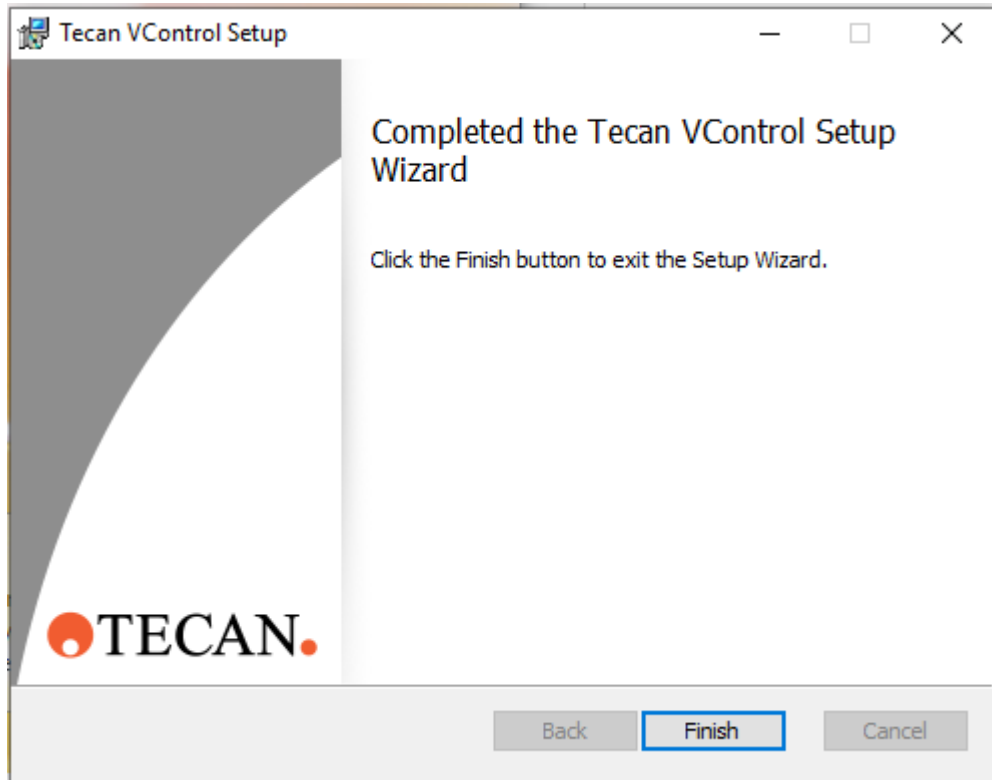
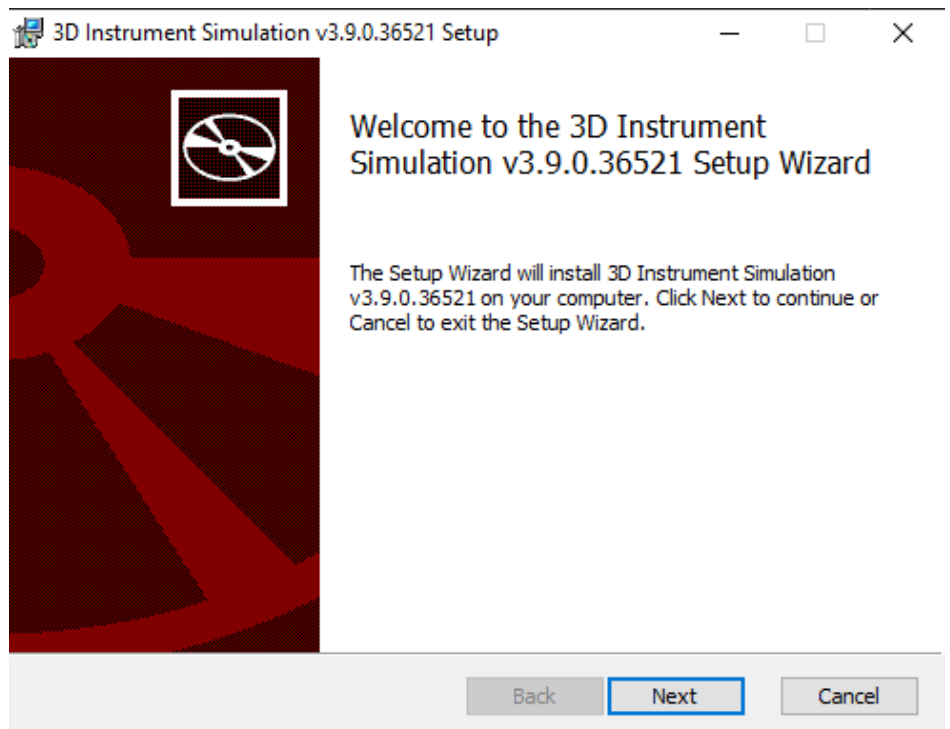
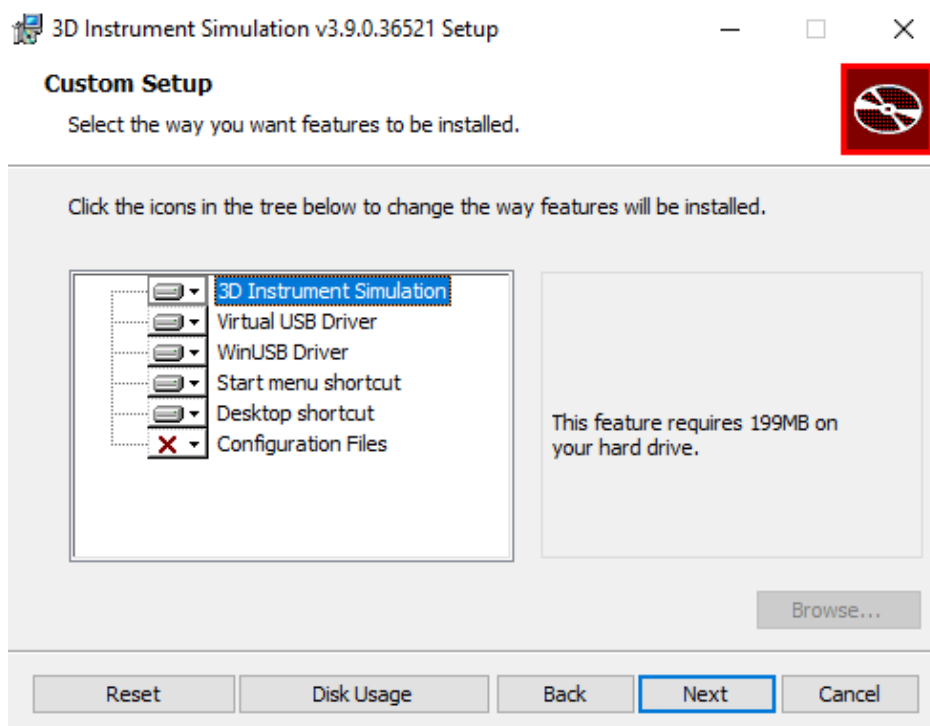
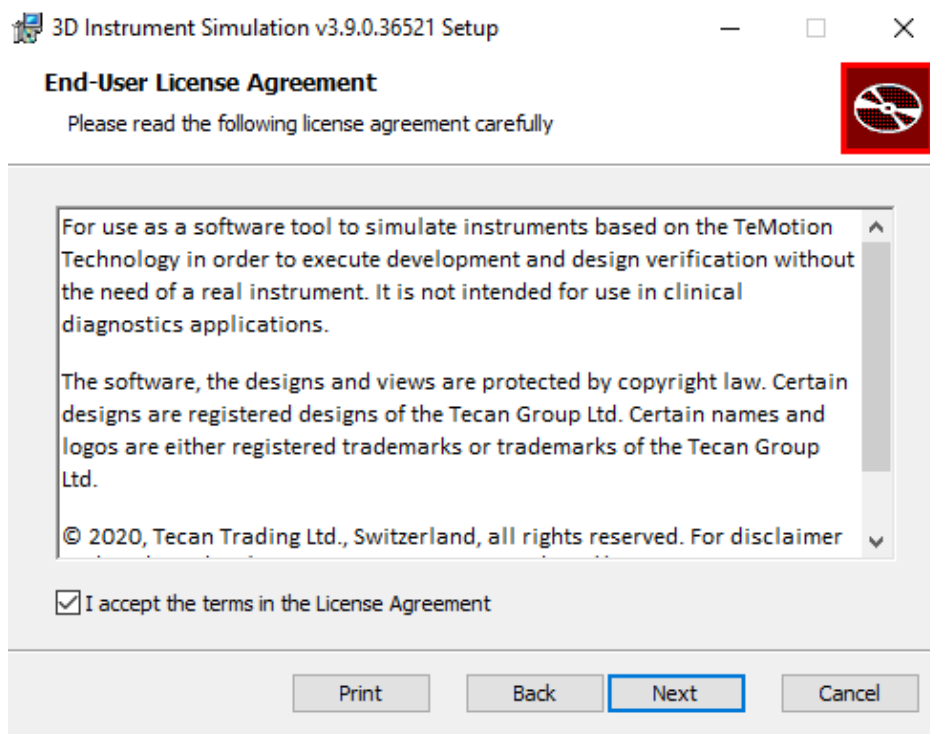


Fig. 7: Installation complete

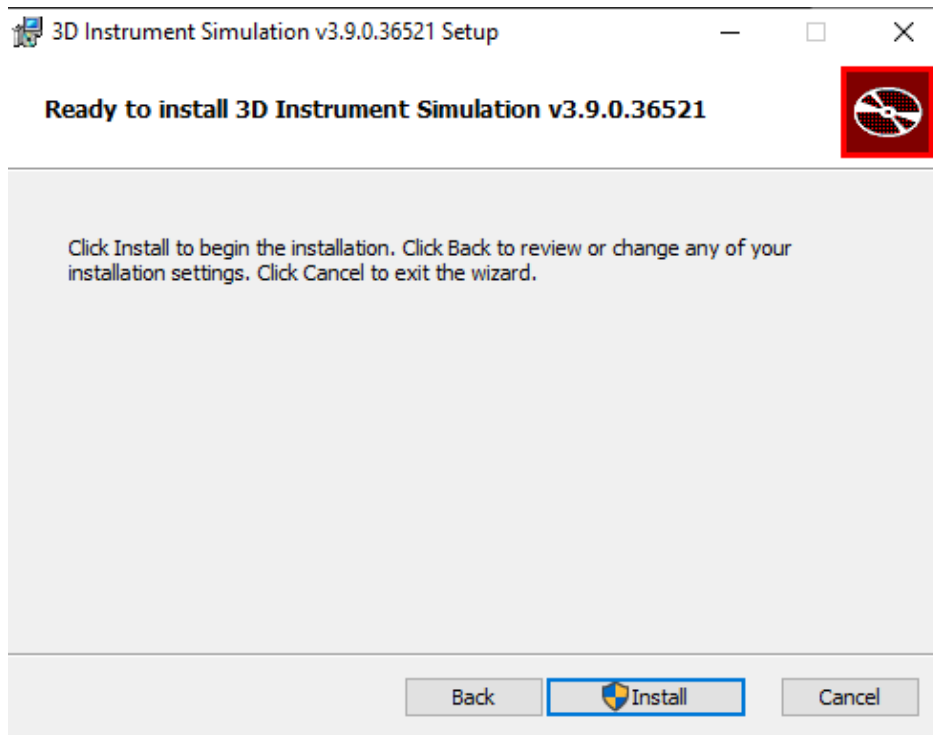
12. Click **Finish** after the installation is finished. The setup wizard screen closes.
13. Click **Step 3: The Tecan 3D Sim Installer Appears.:**



14. Click **Next**.
15. Read and Accept the End User License Agreement, and click Next.



16. Click **Next**



17. Click **Install**

4.3 Activate License

Each instrument is registered on the Tecan Licensing Server with its serial number and the license types that were sold in combination with that instrument. The following two license types are available for vControl: Base license and Gx license. If no existing license is available, you may request your license by typing in your personal data and sending a request to Tecan. The license for vControl is not provided by a dongle or other hardware. The license must be activated on the local PC via the Licensing Client Module. Licenses can be activated online or offline.

Online Activation

1. Start vControl and make sure you are connected to an instrument.
2. In the menu **Settings**, click **Tecan Licensing Client**.
3. The **Licensing Client Module** appears:

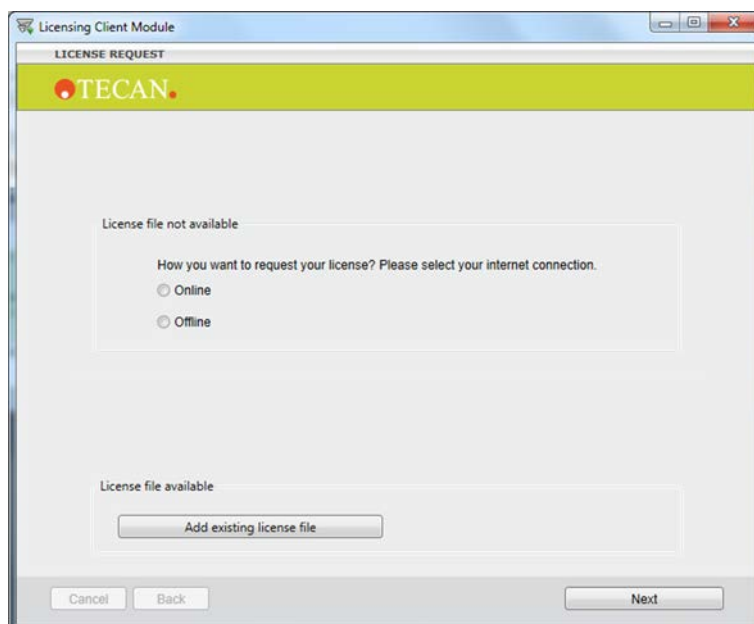


Fig. 8: Add license file

4. Select **Online** and then click **Next**.
5. Leave the **Activation Key** field empty and click **Next**.
6. Fill in the form window accordingly and click **Send Request**. A message appears will appear confirming the successful activation.
7. Click **OK** and restart the vControl. The instrument is ready for work.

Offline Activation

1. In the menu **Settings**, click **Tecan Licensing Client**.
2. Select **Offline** and click **Next**.
3. Choose from the following two modes:
 File Mode: A request file will be generated and this file will be uploaded to the webpage.
 Key Mode: A sequence of characters will be generated and displayed for manual entry on the web page.



Fig. 9: Choose Request

4. Leave the **Activation Key** field empty and click the **Send Request** button. A window with the path to your license request file or your request key will appear.

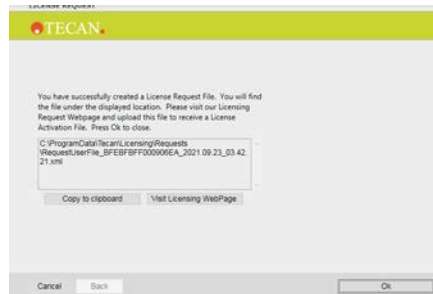


Fig. 10: Licensing Key Path

5. Copy the license request file to an USB storage device.
6. Go to another computer that has internet access to complete the next steps.
7. Open the licensing webpage, which can be accessed by copying the URL below. The URL is also available by clicking the “Visit Licensing Webpage” button in the illustration above.
<https://license.tecan.com/Isprovider/licensingrequest.do>
8. Upload and submit your license request file, or click **Manual Request** to manually enter your request key.
9. A license data file specific for your instrument will be available for download. Copy this file to a USB storage device.
10. Go back to the instrument computer with the copy of your license file.
11. Open the Licensing Client Module and click the **Add existing licensing file** button
12. Select the downloaded license file and click the **Add file** button
13. Restart the vControl application. You can now work with the new, valid license.

Note: After license activation, the current license type is displayed in the bottom left corner of the vControl main window: Simulation mode does not require a license. In simulation mode, the “Demo License” type is always displayed in the bottom right corner of the software. The license file is linked to the serial number of a specific instrument. It is possible to download the license file multiple times. When replacing the instrument computer, create a new license request and download the license file for the new computer.

5 Getting Started

vControl is an application software, which allows the user to program and run fully automated laboratory liquid-handling processes, pipetting and robotic processes on the Veya automation workstation.

The user can simply drag and drop commands by using the 3D editor to create protocols to configure their worktable elements and labware. Finished protocols can be run first in a 3D simulator, without wasting valuable reagents.

5.1 Starting the Software

- The instrument powers on when the application software is started and the connection is established. When not in use the instrument goes in standby mode.
- The system has to be initialized the first time you power it on. Note: To avoid errors first open a worktable in the SW.
- When vControl is started, the TouchTools interface will be shown by default. In case a second screen is connected, both the main software interface and the TouchTools interface windows are shown. Switching between the two windows is possible by clicking the switch button in either window.
- A SW license is required to run the system in real mode. Simulation and script editing as well as System initialization do not require a license. The license must be activated on the local PC via the Licensing Client Module (see Licensing).
- Press F1 to open Software Application Manual.

5.2 Teaching with the MoveTool

The MoveTool is helpful when doing manual teaching of sites or vectors with the RGA for correct labware placement or when teaching labware positioning with the FCA. It is helpful in three ways:

- To check coordinates after moving the arm to a desired position using the Move To buttons in the carrier/labware editor
- To move the arm in certain incremental steps or in a controlled fashion to a desired position using the arrows in the MoveTool window.
- To move the arm in certain incremental steps or in a controlled fashion to a desired position using the arrows in the MoveTool window.
- To move the arm manually to a desired position using the Zero-G mode.

NOTICE

The MoveTool will indicate what tools are mounted on the arms. Tips can be mounted via Direct Commands. Positions will be correctly calculated with mounted tips.

While teaching carrier positions or labware with the doors open, the arms will move at half their normal speed.

- Do not attempt to circumvent the door's safety sensors locks.

5.3 Starting vControl

1. Start Windows.
2. Click **Start > All Programs > Tecan > vControl**.
A startup screen appears during startup.

5.4 Logging In



Contact your vControl administrator if you do not have a user name and password.

1. If User Administration is activated, you must enter your password during log in. For more information on User Administration refer to [“User Administration”](#) [▶ 74].



Fig. 11: vControl Startup Window

2. Click your user name icon.
3. Type your password and press **Enter** or click the large green arrow on the right.



Fig. 12: Password for Administrator

5.5 The Configuration Editor

The Configuration Editor is the main screen and allows a user with Key Operator rights to build a workdeck with all carriers and labware, to script a protocol, to optimize liquid class settings or to teach the sites on the worktable. Simply drag and drop commands and worktable items from the Controlbar (left side) to the main section (right side) to set up a protocol or a worktable.

You interact with the user interface documented in this manual or with **TouchTools**, an intuitive, customizable touch monitor interface. **TouchTools** is now an integral part of vControl. You can quickly create interfaces using **TouchTools** standard features like easy variable entry, or build completely customized interfaces for your operators.

Overview of the Main Window

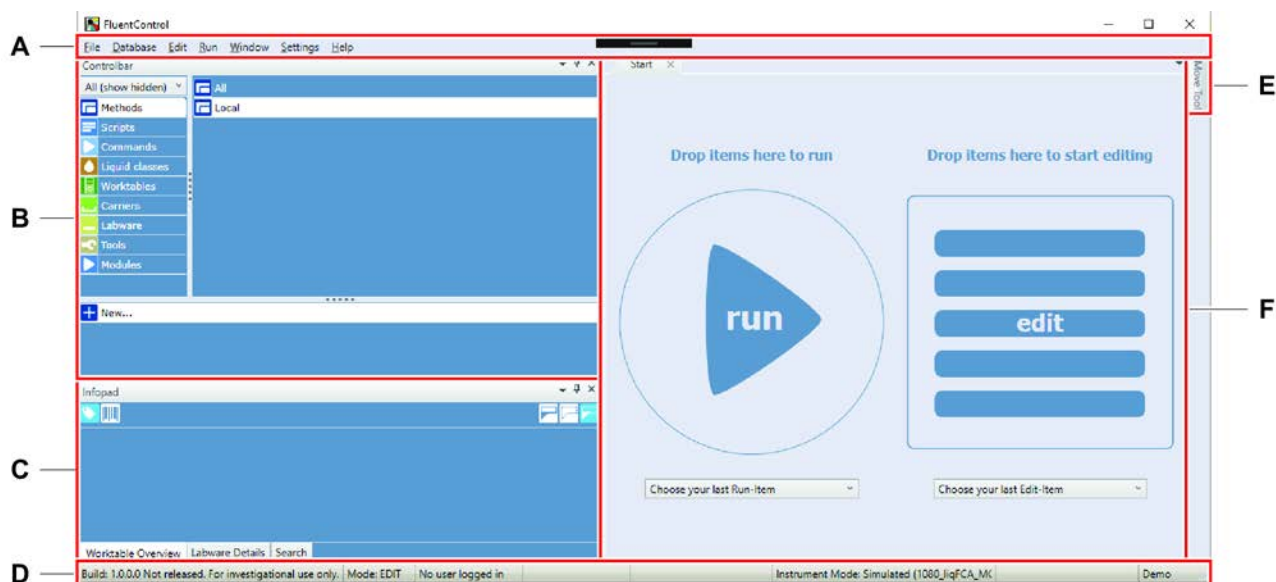


Fig. 13: vControl window on PC-screen

A	Menu bar	B	Controlbar
C	Infopad	D	Status bar
E	MoveTool	F	Main section

Menu bar

The menus are displayed on the menu bar located near the top of the main window. For a description of the **File**, **Database**, **Edit**, **Run**, **Window**, **Settings** and **Help** menu refer to section "[Menus](#)" [92].

Controlbar

The **Controlbar** contains items for methods, scripts, modules, worktables, commands, objects (carriers, labware, and tools), and liquid classes.

Infopad

In the **Infopad**, you see information about objects (carrier, labware, and tools) that are currently selected on the worktable. Alternatively, it shows context check errors that show up during scripting of a protocol. Also there is a pinned "Search" window at the bottom of the Infopad section to be used to find quickly commands and worktable items from either the Controlbar or the Edit window or also it can find comments or key words from an open script.

Start, Section or Main Section

A tabbed page used for quick access to start a run or to start editing.

MoveTool

The MoveTool window is pinned on the right top side of the main window. With the help of the **MoveTool**, a user can move the arms to absolute positions on the worktable for e.g. teaching purposes or in order to check a certain position for correctness. Also the feature Zero-G can be enabled within the **MoveTool**.

Status bar

The area at the bottom of the main window that lists the status of vControl the build number (version), current software mode, instrument mode, and license activated.

5.5.1 Controlbar

The Controlbar is used for selecting objects (**Carriers**, **Labware**, and **Tools**), commands and liquid classes, which are needed to set up a script. The Controlbar is organized in categories. Each category contains a list of objects, which are arranged in groups.

1. Click on the group's title to access group specific items.

The selection of the displayed items can be changed.

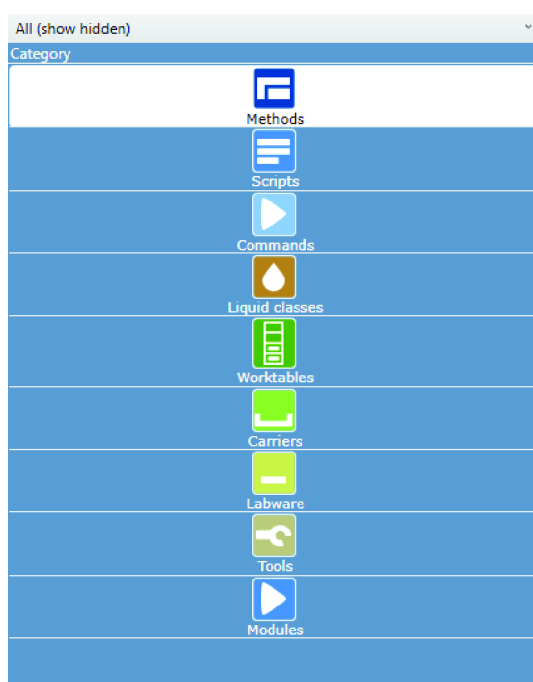


Fig. 14: Controlbar: Category view

2. Click the drop-down arrow to see more options:

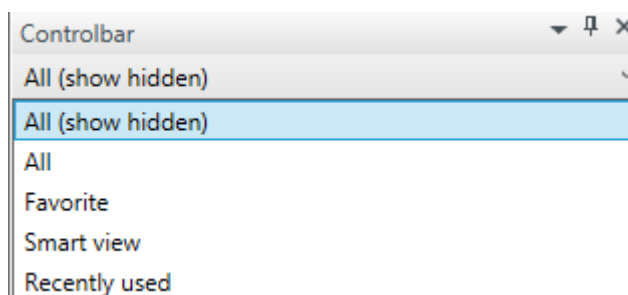


Fig. 15: Controlbar: Selection of displayed items



If you select the Smart view option, for example, only context relevant items are shown in the Controlbar. For the supported views, see also the Controlbar Views command on the Settings menu.

5.5.1.1 Methods

A carrier is a removable support on the instrument's deck. You can use a carrier to hold one or more pieces of labware in place. Carriers can be fitted to the deck at pre-defined grid positions (positioning pins/clips). The pre-defined grid positions are spaced 25.0 mm apart in X-direction. By default, carriers are saved in the base workspace, labware is saved in scripts. Methods can be visible on the touch interface for operators. Methods can also be saved as "Maintenance" or "System Care" to show up under the "System Care" touch monitor option. All scripts in a method must share a base worktable.

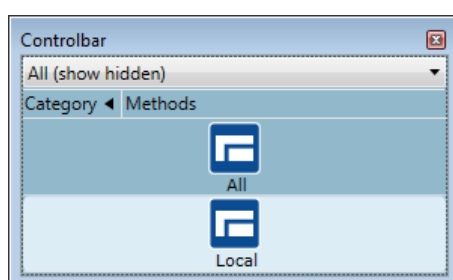


Fig. 16: Methods category in Controlbar

5.5.1.2 Scripts

Scripts are a collection of commands to carry out the actions and interactions required for an assay.

5.5.1.3 Worktables

This editor allows you to edit the base worktable which may be used by one or more scripts.

5.5.1.4 Commands

The **Command** menu containing commands for the installed arms and devices – for example:

- All
- Direct commands
- FCA 1
- RGA 1
- Programming
- Touch Tools
- Worklist
- Sample Tracking
- Worktable
- Communication

5.5.1.5 Liquid Classes

A liquid class defines the robotic actions that will take place during a **pipetting script** command.

Liquid classes are defined per liquid type and pipetting mode (e.g., single pipetting or multi-pipetting, free dispense or wet contact). One liquid class covers the whole volume range for FCA.

vControl comes with pre-defined liquid classes.

Opening the editor for displaying and editing liquid classes

1. To open the liquid class editor, click **Liquid Classes** in the Controlbar.
2. Double-click on any of the liquid classes.

*Alternatively, on the File menu, click **Open > Liquid Class**.*

5.5.1.6 Carriers

A carrier is a removable support on the instrument's deck. You can use a carrier to hold one or more pieces of labware in place. Carriers can be fitted to the deck at pre-defined grid positions (positioning pins/clips). The pre-defined grid positions are spaced 25.0 mm apart in X-direction. By default, carriers are saved in the base workspace, labware is saved in scripts.

With vControl, you define an object (e.g., a microplate) only once, and then, you use it at any required position on the worktable.

Coordinates

vControl uses a hierarchical system for the coordinates:

- The labware coordinates are relative to the site on which the labware is placed.
- The site coordinates are relative to the carrier.
- The carrier offsets are relative to the grid position on the worktable
- The grid positions on the worktable have a spacing of 25 mm.

This hierarchy allows you to place the labware at any suitable site, carrier and worktable position without having to redefine its coordinates.

Types of carriers

The **Carriers** menu contains the following types of carriers:

- Deck Segment, which attach to a grid and provide nests for labware
- Grid Segment, which attach to the grids and provide pins for runners to slide on and to position hotels
- Runner, which slide on and off grid segments (e.g., a tube runner)
- Hotel
- Nest, the part of the carrier that provides a position for a microplate or SBS format tip box
- Device
- Deck Extension

5.5.1.7 Labware

Labware refers to consumable items such as microplates or sample tubes. Microplates have a regular arrangement of containers or cavities (wells) for holding liquids or samples.

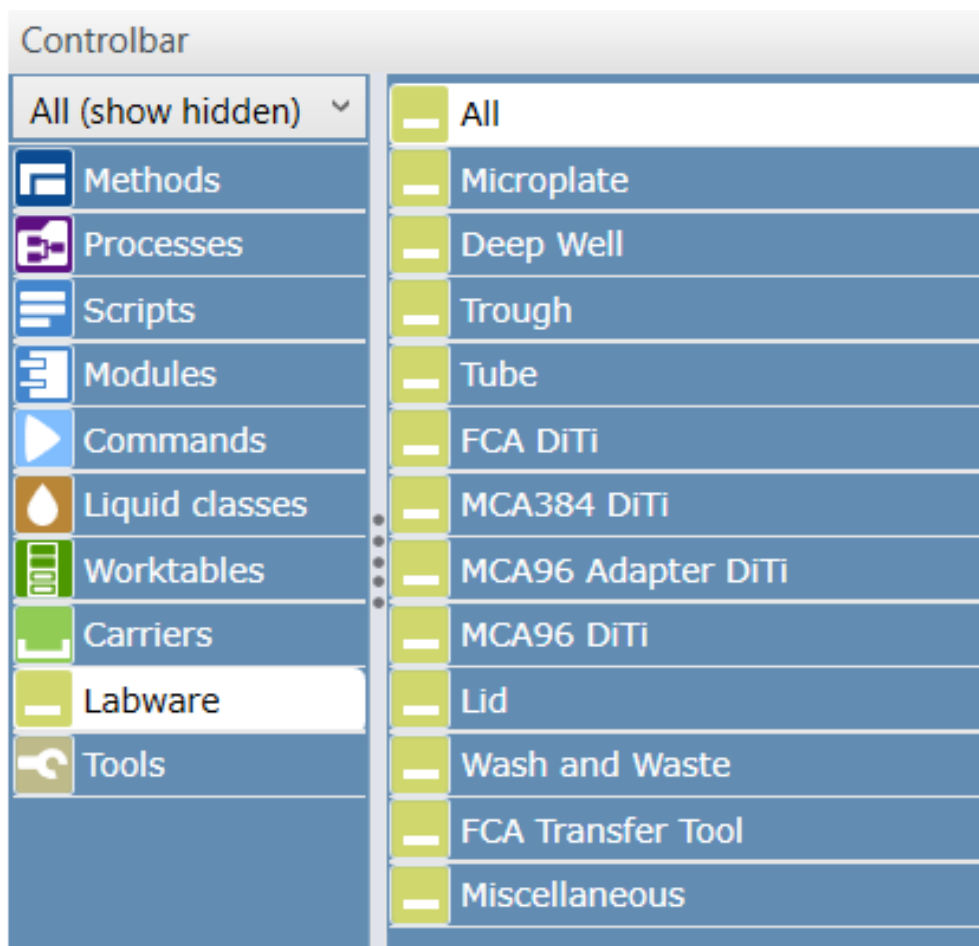


Fig. 17: Labware Tab

The **Labware** menu contains the following sub-categories:

- Deep well Microplates.
- Disposable Tips used only on the FCA arms.
- Lids on Microplates and Lids on Disposable Tip Boxes.
- Flat Microplates.
- Troughs for the FCA arms and SLAS format troughs.
- Tubes with or without lid of different sizes.
- Wash stations and liquid waste stations.
- FCA Transfer Tool: Disposable to dispose nested DiTi layers.
- Miscellaneous: E.g. spacers.

Allowed Labware

If you have already put a carrier onto the Worktable Editor, click on it to see which labware types are compatible with the currently selected carrier.

The vControl database already contains a large selection of pre-defined labware types. You only need to create or edit labware if you cannot find a standard item which is suitable for your needs.

5.5.1.8 Tools

Tools are objects which attach to arms and are not thrown away, for example **Fixed tips, or RGA fingers.**

5.5.1.9 Modules

Modules are specially designated scripts that can be inserted into the other scripts.

To add a Module to a script, drag the Module from the Module section of the Controlbar into the Script Editor. Module commands are like synchronous Subroutine commands. To run a Module asynchronously, call it using a Subroutine command:

Transfer values to module: If the Module contains variables with the scope Parameter, they will be automatically inserted into the "Transfer values to module" section of the Module command after it is added to the script. Only variables with scope Parameter are shown in the "Transfer values to module" section. It is not possible to add additional variables to the list to pass them to the Module.

- **Value from main script:** This list contains the default values of variables with scope Parameter in the Module. The values can be modified and will then be used by the Module. If the value of a variable from the main script shall be passed to the Module, enter the variable name in this field instead. Values for variables of type String are wrapped in double quotes. Empty string values will lead to context check errors.
- **Variable in subroutine:** This list contains all variables with scope Parameter used in the Module. These fields cannot be edited.
- **Transfer variable values from module to main script:** Variables whose values shall be transferred from the Module to the main script after the Module has been executed can be manually entered here.

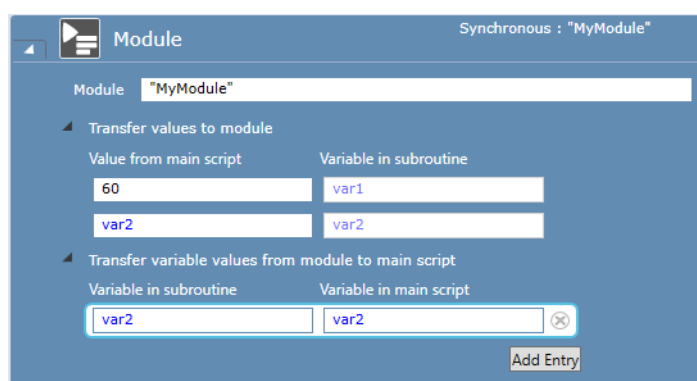


Fig. 18: Module command

5.5.2 Infopad

In the Infopad, you see information about carrier, labware and tools that are selected on the worktable.

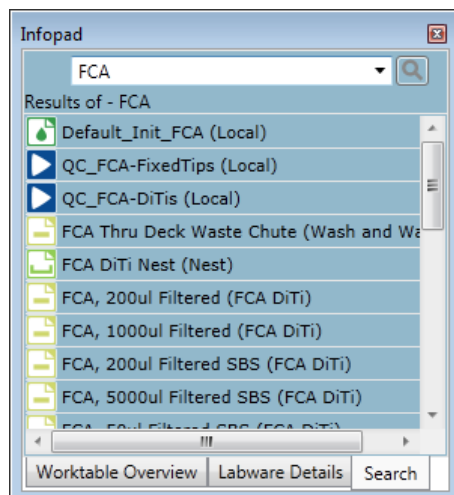


Fig. 19: Search example in Infopad

Errors

Context check errors are displayed in the error tab of the Infopad. It shows the errors of the active editor and is labeled with the name of the current opened object. Click on a line in the error tab of the Infopad to select the corresponding line in the script editor.

5.5.2.1 Worktable Overview

The worktable overview can be easily navigated by clicking on the worktable. It is where you can rename labware and hide or reveal labels.

5.5.2.2 Labware Detail

Labware Details will show a graphic of the selected part, the name, label, and additional information.

5.5.2.3 Search

You can search for objects against labware name and part number. The results are shown in the Infopad area. The result entries have the same drag and drop behavior as in the Controlbar. The search covers the currently opened script editors and method editors as well as all Controlbar elements. You can also search for opened script content (e.g., variables).

5.5.3 Start

In this pane, also called the main section, you find the run and the edit part. By drag and drop to these sections, you can **Edit** or **Run** a script.

From the vControl Start pane, you can perform the following tasks:

- **Run:** Run an editor item like a script/method or a direct command.
- **Edit:** Edit an editor item.

Two drop zones of Start: Run and Edit

The Start pane provides two drop zones, which enable the user to run or edit an editor item like a script/method or a direct command. The start of the drag and drop operation is the search and the Controlbar and the drop target is one zone for edit, the other for run direct.

The Start pane has two recent lists, which provide last ran scripts and methods, and a second list, which provides entries for create editor items for script and method and a list of last edited items.

5.5.4 MoveTool

After the initialization of the instrument, the arms cannot be moved without vControl **MoveTool** support. If vControl is connected to an instrument (real or simulated), you can use the **MoveTool** and move the arms individually by clicking the arrows in the **MoveTool** pane. The arms speed up upon continuous movements.

5.5.4.1 MoveTool Pane

If the instrument is switched to the Zero-G state, then you can move all axis of all the connected arms manually.

CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

5.5.4.2 Teaching and the MoveTool

If you are teaching a custom carrier, labware or vector, there is a direct connection between the corresponding editor and the **MoveTool**:

- When you click **Move** in one of the editors, the robotic arm selected in the **MoveTool** will move to the desired position.
- Transfer will record the actual position of the arm selected in the **MoveTool** in the current field.
- Show Delta will show the difference between the taught position and the current position of the arm selected in the **MoveTool**. It will not change the definition of anything.
- Adjust Site will use the labware or vector definition and the current position of the arm selected in the **MoveTool** to calculate new site offsets for only the topmost carrier.

The **MoveTool** will indicate what tools are mounted on the arms. Tips can be mounted via **Direct Commands**. Positions will be correctly calculated with mounted tips.

NOTICE

Transfer will take the position of the arm, even while the arm is in Zero-G.

In this version of the software, **Show Delta** and **Adjust Site** will only work if the arms are active, i.e. on standby.

- To work around this issue, take the arm out of Zero-G before using **Show Delta** or **Adjust Site** by clicking on a directional button or the Zero-G button in the Move Tool.

While teaching carrier positions or labware with the doors open, the arms will move at half their normal speed. Do not attempt to circumvent the door sensors.

5.5.4.3 Home Position

A home position is defined for each robotic arm. After initialization of the instrument, the robotic arms move to their home positions. The default home position of the FCA module, for example, is over the nearest wash/waste station with Z-axes on Z-travel position. To configure home positions, use the Configure System command on the **Settings** menu, expand the devices section and click **Home-positions**.

5.5.4.4 Zero-G

To release the resistance on the motors to allow the robotic arms to be positioned gently by hand, click **Zero-G**. Click **Zero-G** again to reset the instrument on Standby. Zero-G mode will be canceled automatically when a predefined interval of time has passed without further user interaction.

5.5.4.5 Moving the FCA

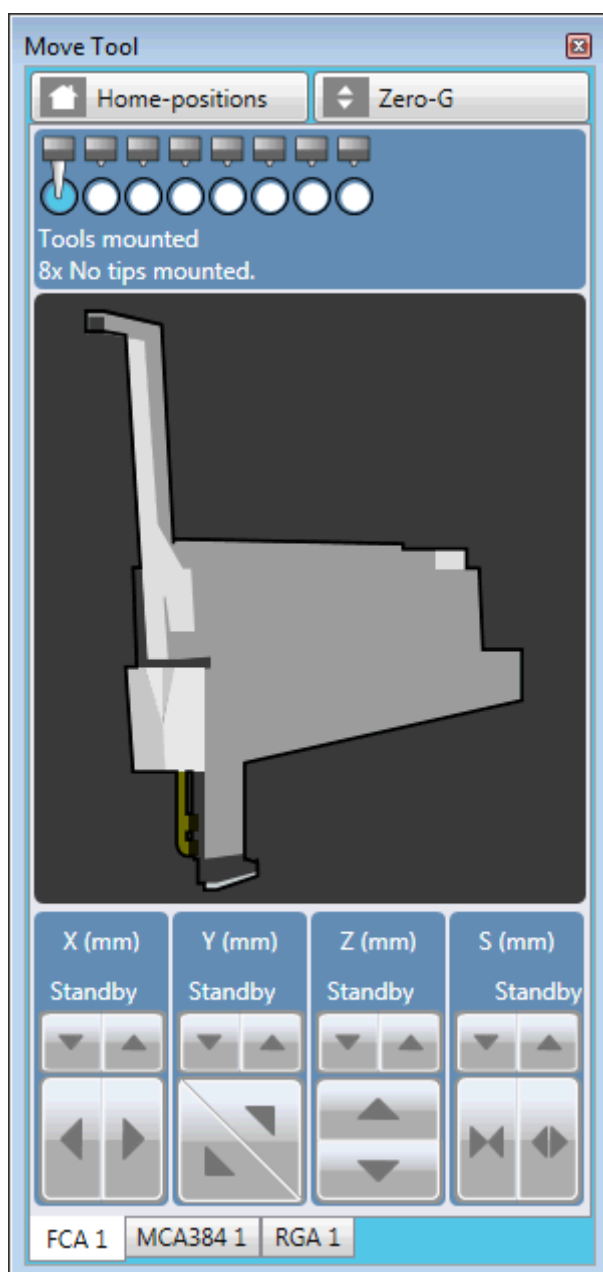


Fig. 20: MoveTool: FCA

X [mm]: Movement axes from left to right: Use the Left Arrow and Right Arrow keys on the numeric keypad or click the appropriate button.

Y [mm]: Movement axes from rear to front: Use the Up Arrow and Down Arrow keys on the numeric keypad or click the appropriate button.

Z [mm]: Movement axes from top to bottom

S [mm]: Spreading of the FCA channels (expand or narrow)

To move the arm one step (0.1 mm), press the appropriate arrow key. If the key is kept pressed the arm is continuously moved in the direction as defined in the table below:

<div>7</div> <div>expand (+S)</div>	<div>8</div> <div>rear (+Y)</div>	<div>9</div> <div>up (+Z)</div>
<div>4</div> <div>left (-X)</div>	<div>5</div>	<div>6</div> <div>right (+X)</div>
<div>1</div> <div>narrow (-S)</div>	<div>2</div> <div>front (-Y)</div>	<div>3</div> <div>down (-Z)</div>

5.5.4.6 Moving the RGA

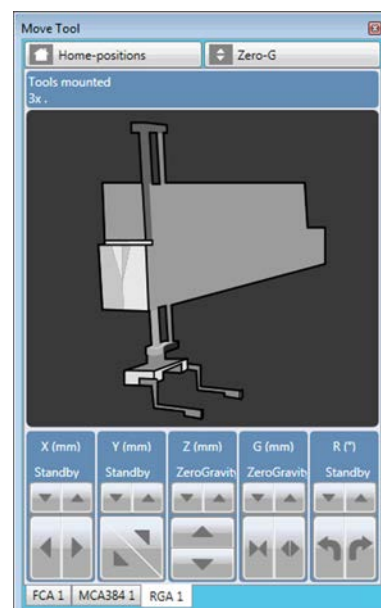











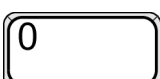

Fig. 21: MoveTool: RGA

X, Y, and Z as described above.

G [mm]: From narrow to wide gripper distance. Use the 0 and . keys on the numeric keypad as described in the table below or click the appropriate button.

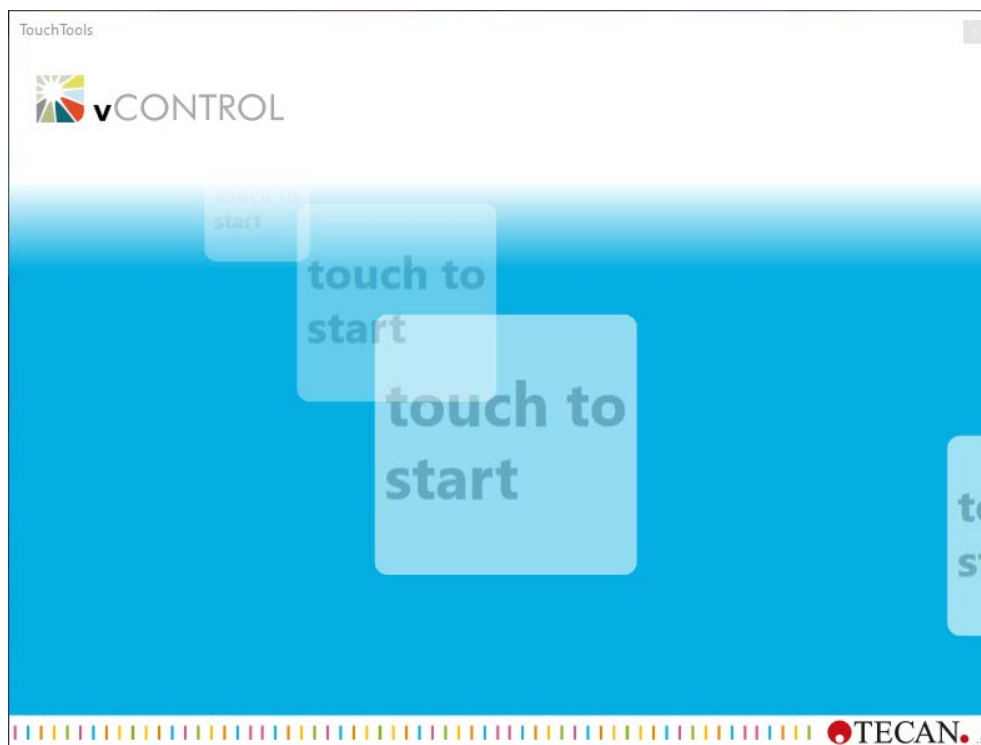
R [°]: Clockwise rotation of the gripper when viewed from above. Gripper rotation is measured in 0.1° steps.

Use the 7 and 1 keys on the numeric keypad as described in the table below or click the appropriate button.

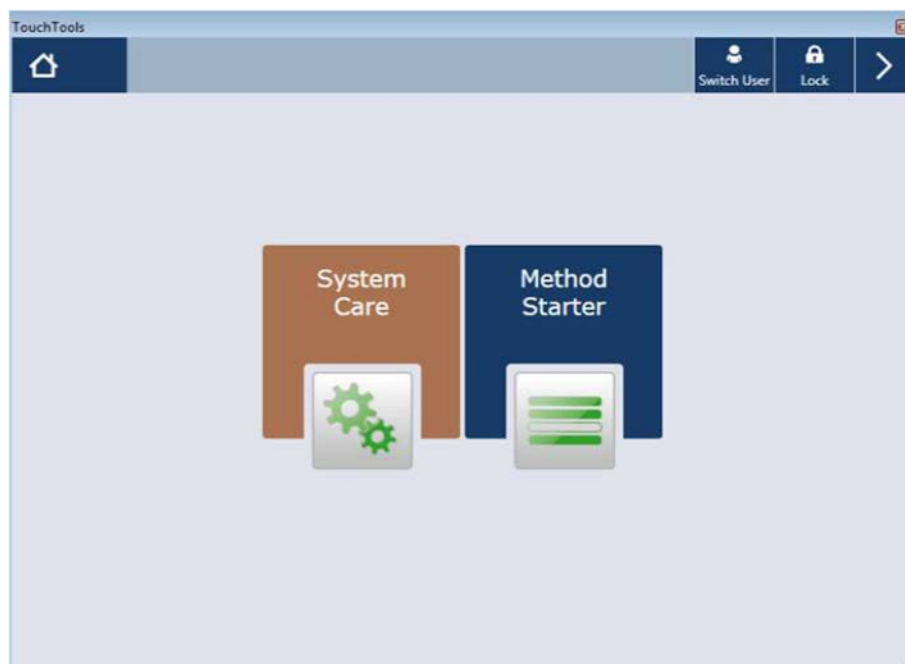
 clockwise (+R)	 rear (+Y)	 up (+Z)
 left (-X)	 	 right (+X)
 counterclockwise (-R)	 front (-Y)	 down (-Z)
 narrow (-G)	 wide (+G)	

5.6 TouchTools

The TouchTools window provides an intuitive interface for Operators to run daily workflows. This can be validated methods or maintenance methods. A loading guide can be setup and made visible on the Touchtools interface to guide the user through the different loading steps. Furthermore the operator can enter input, confirm actions and continue to the next action through this interface.



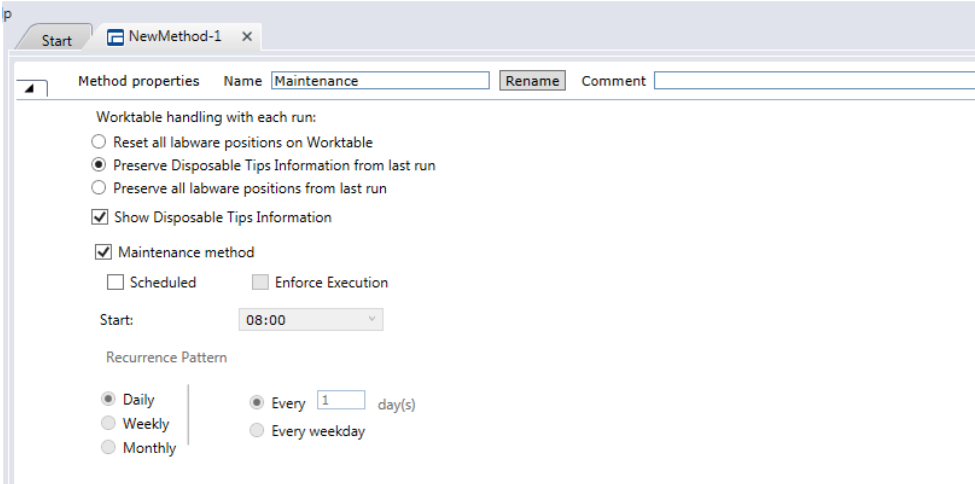
To start TouchTools, touch the screen. After startup, the home screen will show the following categories:



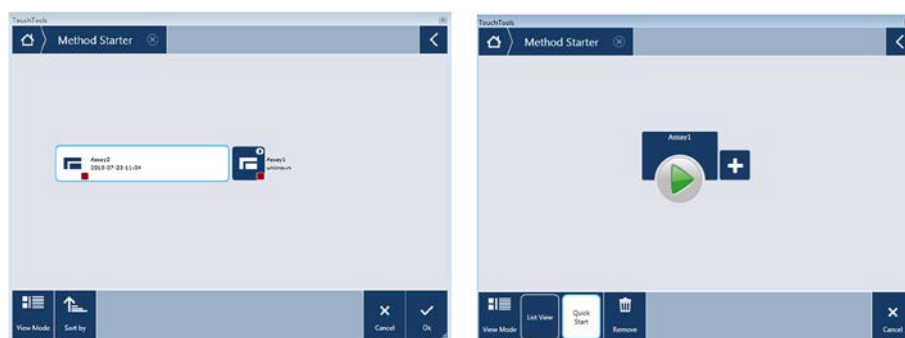
System Care: Select & start maintenance methods. Note: If a system care method is due then the System Care icon will flash orange.

Method Starter: Opens the menu where the application method to run can be selected.

Only methods can be run from the TouchTools interface. To make a script available on TouchTools it must be prepared and saved as a method. However, not all methods will automatically be shown on the TouchTools, only those with the option "Is visible in TouchTools". Check the option "Maintenance Method" if you want a Method to show up under System Care.



In the Method Starter, you can switch between the Quick View and the List View. The List View will show you all methods made available on TouchTools, whereas the Quick View allows the user to define favorites which can then be started more easily.



If you want to delete favorite methods from the Quick View, press the Remove button and select the method to be removed.

It is also possible, at any time, to switch users or lock vControl from TouchTools; to make those options available expand the tab by clicking on the arrow on the top right corner in TouchTools. To switch to the main editor of the application click the button "Editor" in the expanded tab.

Editor Button

The **Editor Button** in Touch Tools application provides a functionality to easily switch from TouchTools to the main editor window.



If only one physical screen is attached to the instrument the default window is always the used window before closing the software. A switch button in both windows allows to switch between the windows if necessary.

5.7 Scripts

5.7.1 Good to Know

- Script content gets validated during edit time. If there is non-valid input or missing fields or actions that can't be executed this is shown as Context Check error in the Infopad.
- Flags: Script line flags allow you to quickly deactivate script lines or set breakpoints.
- Details: for every command, parameters can be shown or hidden.
- Scripts can be run exclusively or together with other scripts as a method.
- In a method, you can specify how many times each script is run.
- Ability to preserve Labware positions from script to script using a method.
- Step-by-Step mode pauses before each step in a run simplifying development.

5.7.2 How to Create a new Protocol

5.7.2.1 Create a new Worktable layout with the worktable editor

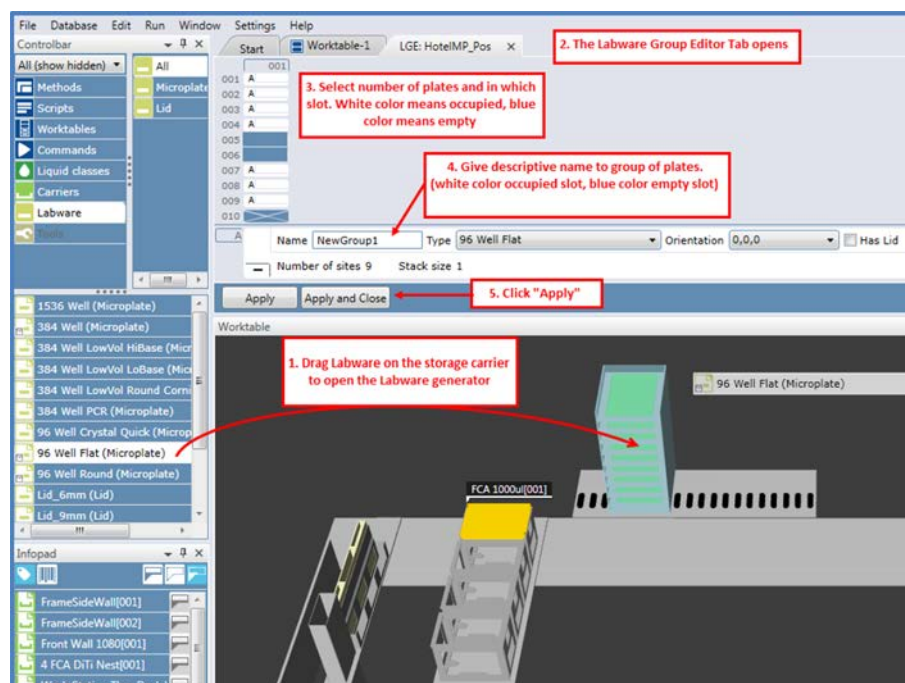
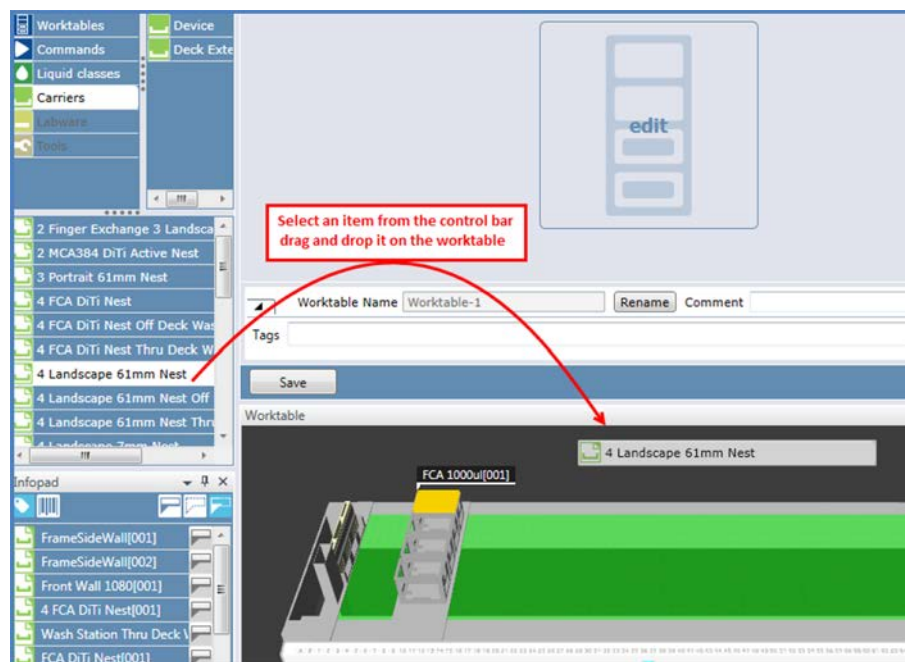
The first step in the process of setting up a protocol is to create the worktable layout that will be linked to the script and will be used to program the sequence of commands. To setup the base worktable use the Worktable Editor. Later, script-specific worktables can be extended and adjusted within the worktable window in the Script Editor.

If a base worktable representing the physical worktable is already available choose the worktable from within the script editor.

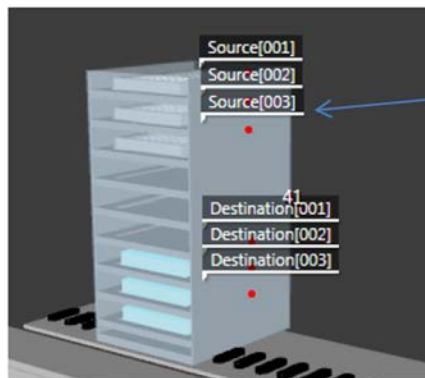
In the worktable editor you see a graphical representation of the pipetting instrument's worktable. To create a new worktable, select "new" in the control bar under category Worktable; a worktable editor tab will open.

Add elements on the so called base worktable that are static (cannot or will not be moved) like segments and nest or tip waste station, devices or static labware like liquid waste or some troughs. To place dynamic objects onto the worktable like plates, tubes or tip boxes use the worktable window in the Script Editor. (Add info

here): Alternatively, labware can also be placed in this Editor and then later be moved to the script via right -click on the labware -> Move to Script.



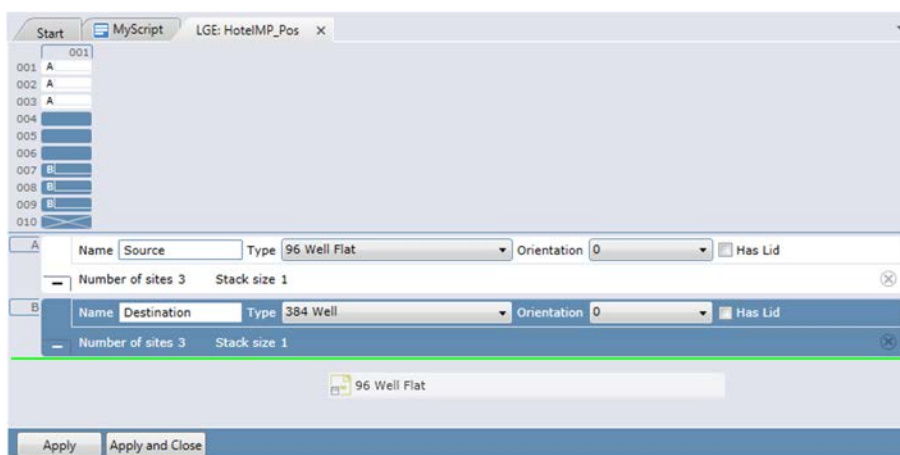
In the Labware Group Editor single or multiple groups of Labware can be created.



Labware Group name:

A group of plates defined in the Labware generator of FluentControl. The group name, type, quantity and storage location are defined. The group name is usually a descriptive name. You might have several different groups of plates consisting of plates of the same Labware type

To create a second group of plates just drag & drop a new Labware below the current group.

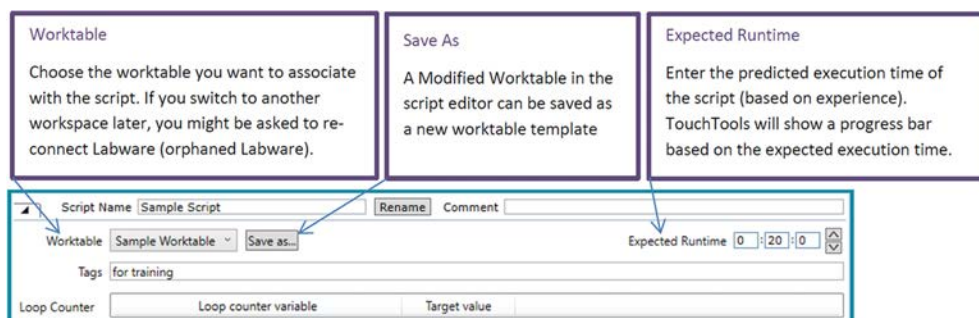


With all elements added to the worktable, save it. All worktables saved are listed in the homonym category in the control bar.

5.7.2.2 Create a Pipetting Script in the Script Editor

Open a new script editor (double click “new” in the control under the category Scripts; a new tab will open on the main window).

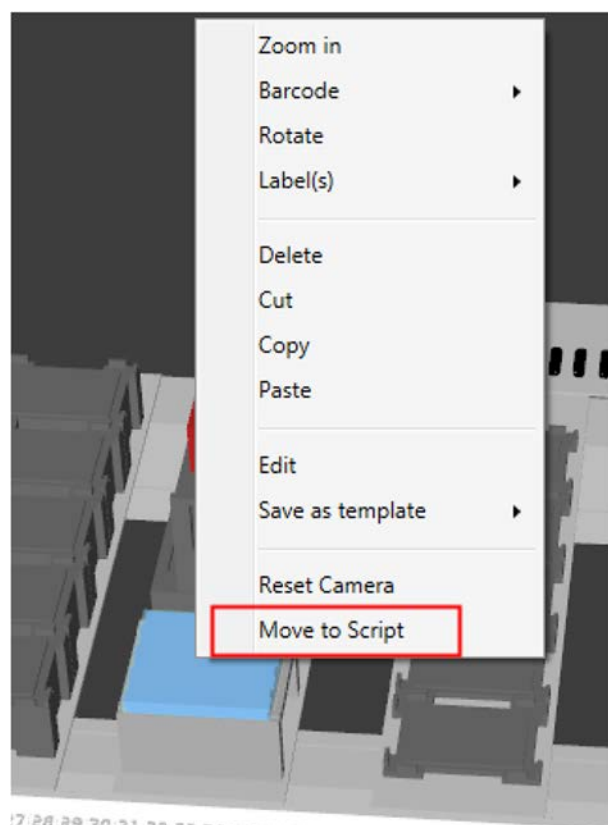
Link one of the existing worktables to the new script. Expand the Script Name section at the bottom of the script editor and select the worktable from the drop-down list.



First, place remaining needed labware like plates, tubes or missing troughs onto the worktable via drag&drop from the **Controlbar**. For labware groups like plates in a storage device or tubes in a tube runner use the labware group editor to define quantity, exact location and orientation. The labware group editor is automatically opened when placing the respective labware into a storage device or tube runner. (Make an info here): Alternatively, for any given carrier or nest that doesn't have this functionality by default it can be defined via Custom Attributes settings whether the **LabwareGenerator** shall be opened or not.



Modifications of the worktable in the script editor do also change the original worktable; however this is only true for objects which are flagged "Base Worktable" only; if the object is flagged "Script" it is only saved together with the worktable linked to the script. You can move an object from script to base worktable level and vice versa by right clicking the object and selecting the option "Move to Base Worktable" or "Move to Script", respectively. Default, segments.at script level (Moving Objects in between Worktables)and nests are at base worktable level whereas Labware are at script level.



The Script Editor is used to set up a script. Commands for all devices are available in the Script editor.

After the worktable is setup, the protocol or Script can be created. For this drag&drop the respective "Commands" from the Controlbar into the empty edit field one by one. The sequence of execution will be from top to bottom generally in a sequential linear manner. More advanced scripts contain loops or If-else

statements or even subroutines that can be executed asynchronously, i.e. in parallel. The commands are colour-coded and categorized based on hardware. .

1. Add Labware to the worktable that are specific for the script

2. Create Pipetting Script. Adds commands from the Controlbar

Wash tips of the Liquid Handling arm

Transfer plate from Hotel to Pipetting nest

Sample are Transferred from Tubes to source plate

Plate is transferred back to the Hotel

5.7.2.3 Change the worktable

Adapting a script to run with another worktable can be done in a few steps. This facilitates sharing of scripts on different machines. In order to switch the worktable your script runs with, simply choose another worktable in the script editor. The software will re-assign the commands in the script to a corresponding Labware found on the new worktable. If there is Labware that is associated with the Script (not part of base worktable) missing on the new worktable and it can't be placed

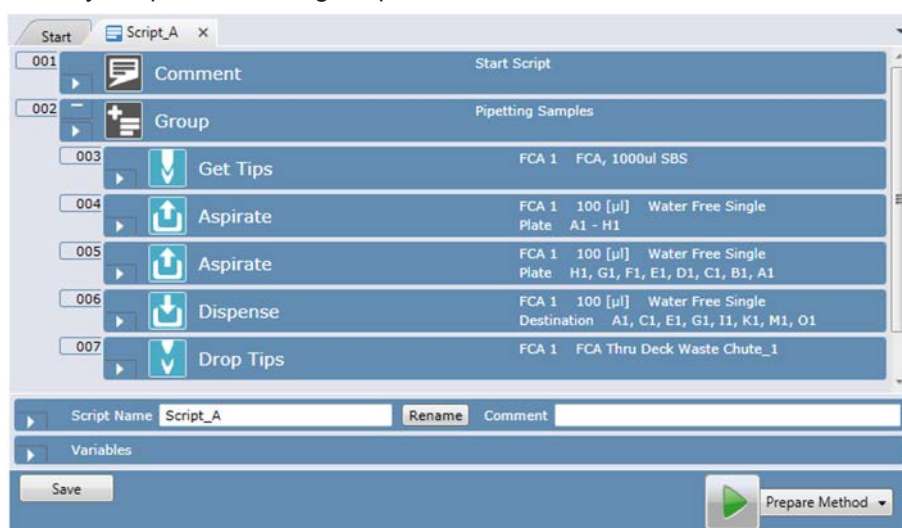
automatically onto the new worktable for any reason, it will appear in the list of Orphaned Labware. Simply drag and drop orphaned Labware to the new worktable to re-connect them.



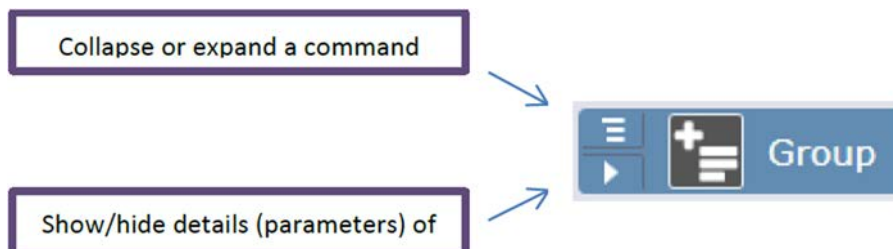
Fig. 22: Orphaned Labware

5.7.3 How to Edit a Script

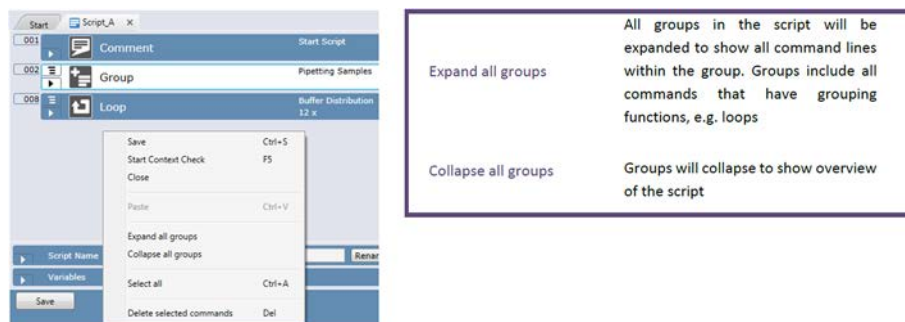
When you open an existing script, it will look like this:



Individual command lines can be expanded/collapsed. This allows you to hide details for a better overview.



By right-clicking outside command lines you get a window that allows you to expand/collapse all groups in the open script:



By right-clicking on a command line a window opens with options to modify the specific command line with options to delete, copy & paste, disable it or set a breakpoint.

Disabled command



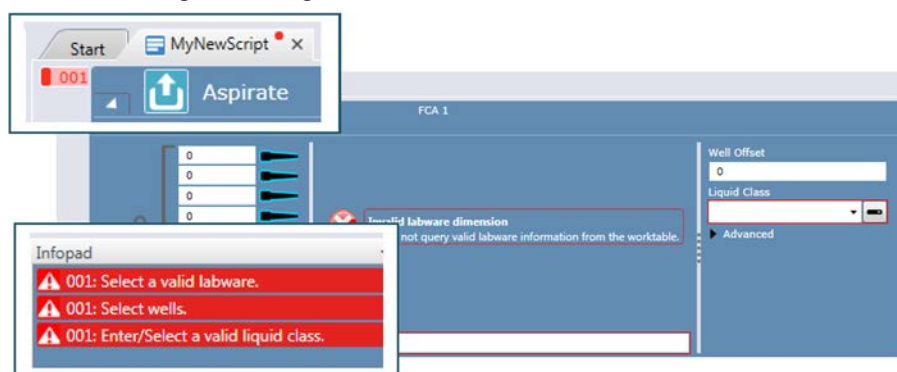
Breakpoint



Context Check

A command marked red indicates there is a problem or error in the script and hence can't be executed. Details about the error are visible in the infopad. For more details on the so -called "Context check errors" are described in chapter 8.4.3 (please make reference to this chapter here).

Auto-context check can be disabled from the Run menu. If the script contains errors, the faulty command(s) are not marked red and you can execute a script after confirming a warning that it was not verified.



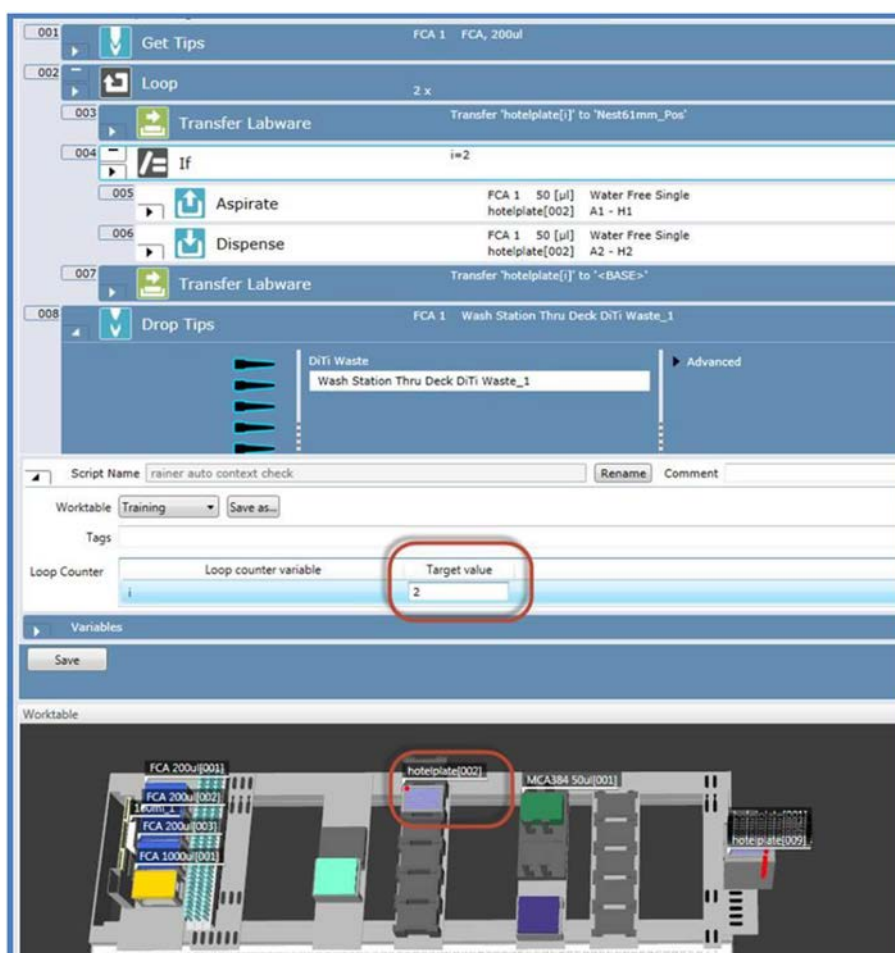
If the script contains conditional jumps or branches, context check only reviews the parts of the script which are executed with the default values of the variables. Target script lines that could not be context-checked will be made visible with a yellow flag and a warning in the Infopad. The worktable will only show a warning.

The sections of the script skipped by context check have to be separately reviewed. To accomplish this, temporarily modify the values of the variables to ensure all script branches are verified.



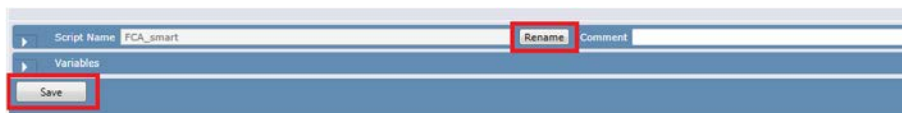
Please note that if the non printing characters (ASCII characters from 0 to 31 except 9, 10 and 13) are used in the input fields there will be no context check error shown. The non printing characters will be ignored and they will not be saved.

The example below shows a loop with loop counter “i” where the worktable is simulated for the second cycle of the loop.

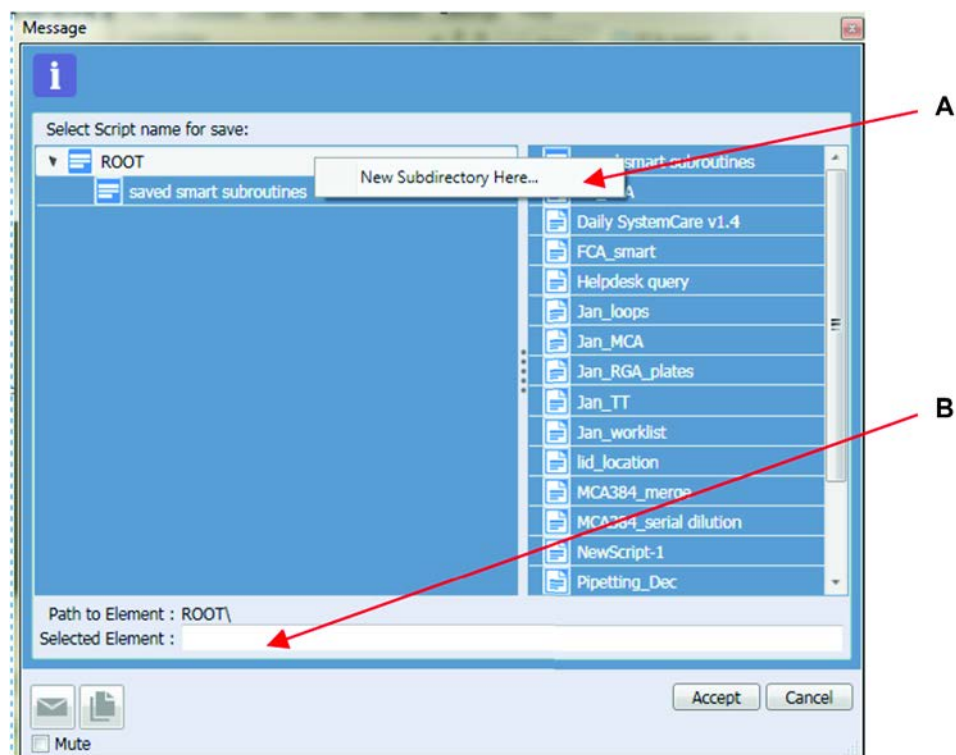


5.7.4 How to Save a Script

Once you are done editing the script, it is necessary to save it. You may rename the script by selecting 'rename' which will allow you to then modify the 'script name field'. Then select save.



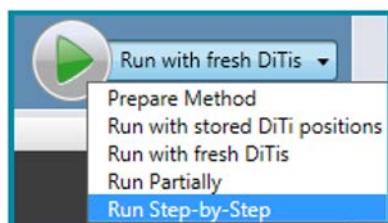
In the case you wish to make a copy of the existing script, choosing 'rename' will only overwrite the name of the current script. It does not create a second instance! To create another script select from the File menu 'save as'. The below user message box will appear:



Location	Description
A	If you wish to save the script to a subdirectory of the root folder, right click on the 'root' header to have the 'new subdirectory here' option populate.
B	Write the new desired script name in the Selected Element box.

5.7.5 How to Run a Script

Scripts can be run independently or combined into a method.



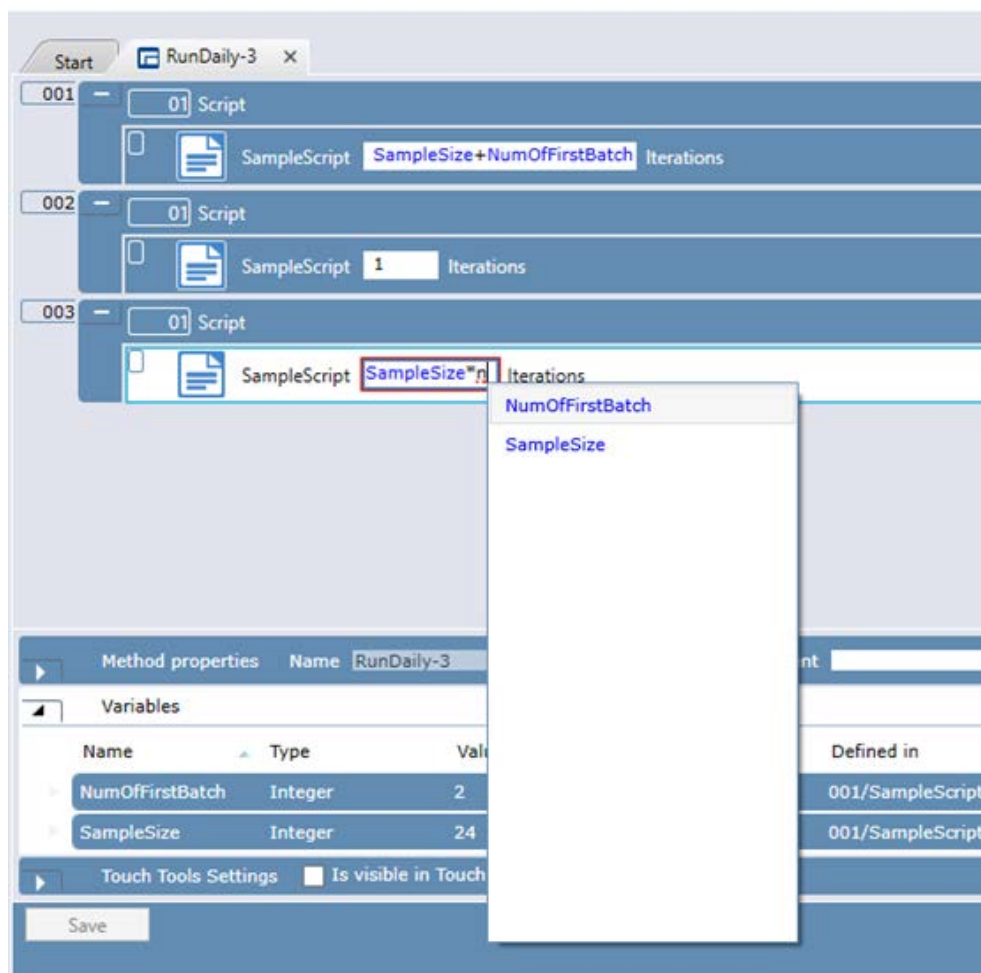
With this option you are shown the method editor tab.

A method consists of one or several different scripts using the same worktable. For every script you can specify how many times it is run (number of iterations).

The method editor offers additional settings: “Reset all labware positions on Worktable”: After the method is finished the Labware (plate and DiTi) status will be reset. “Preserve Disposable Tips Information from last run”: The plate positions will be reset but the DiTi status will be stored. With the option “Preserve all Labware positions from last run”; vControl will update the worktable with Labware information after the run. Labware information includes 1) Preserve the last DiTi position and continue from there the next time a script is started; 2) Update Labware positions on the worktable to the current locations after the run, and 3) Preserve Labware barcodes.

You can make the method visible on TouchTools. Note: individual scripts cannot be made visible on TouchTools, only methods, and by default not all methods will appear on TouchTools.

In the Variable section you will find the list of all variables defined in the individual scripts with scope run and iteration.



The number of iterations can be defined as an integer, a variable with the scope Run or an expression. If a variable or expression is used, the number of iterations to be executed is calculated immediately before the batch is started. If the calculated value is not an integer, the default value '-1' is used. The Floor() and Round() expressions can be used to ensure that an integer value is used. The trace view and audit trail record the number of iterations started for each batch in the method.

Run a Script

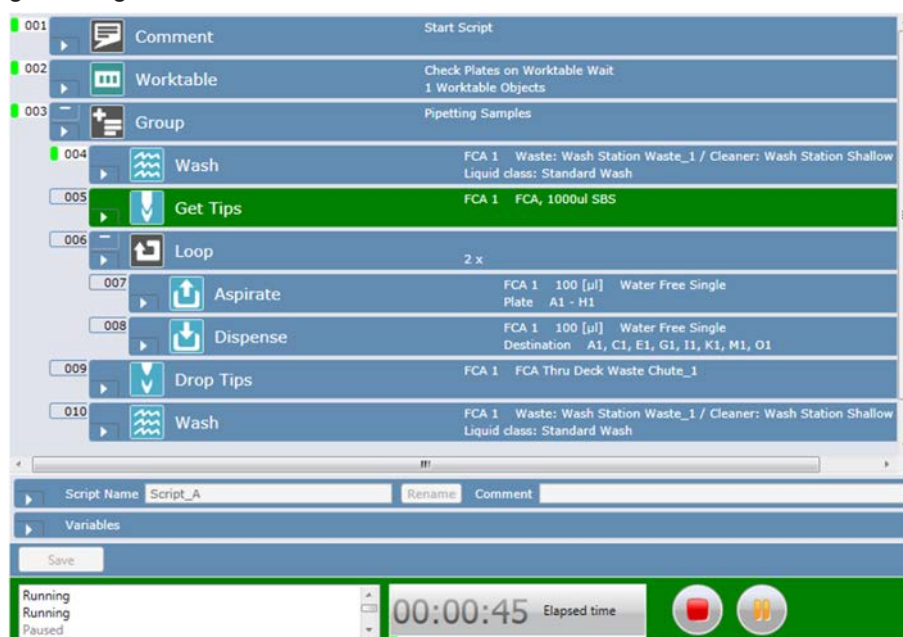
In contrast to running a method, running a script will execute a script directly, and just once.

- **Run Partially:** executes the current selected script lines. To specify the command lines to run, highlight them in the script editor
- **Run with stored DiTi positions:** resets all Labware positions on the worktable but preserves DiTi information from last run, including the positions of tip trays, boxes, and layers in nested tip stacks.
The DiTi position is preserved between different scripts if they use the same Worktable or if the Labware label is the same.

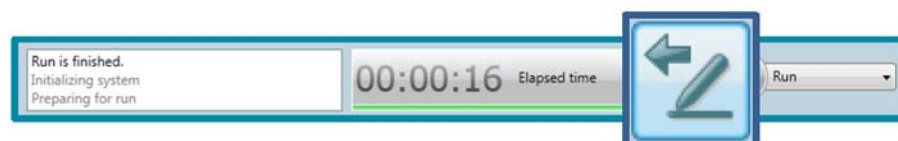
- **Run with fresh DiTis:** resets all Labware positions on the worktable and stored disposable tip information from the last run. DiTis are picked up from the first position in the selected DiTi box assuming that the box is full.
- **Run Step-by-Step:** the execution of the script is paused before each step.

Running

When a method is run, the currently running script is shown. The current step being executed is highlighted in green, whereas already executed steps carry a green flag.



After a run is finished switch back to the edit mode by clicking the “Switch to edit mode” button:



Active Stop & Resume

Opening the door or pressing the stop button will trigger an ActiveStop. The system will stop as fast as possible. Some actions will be allowed to finish (e.g. Pickup Tips, Drop Head Adapter). Most commands will stop immediately, including Aspirate & Dispense commands. If a run has been interrupted by an ActiveStop, the run will be “Completed with Warnings” as incubation times or pipetting steps may be influenced. Reports will also indicate which pipetting steps were interrupted.

After the Arms have stopped, they will enter Zero-G mode.

Upon pressing retry or continue, the system will ensure the doors are closed and will resume whatever steps were interrupted or continue with the next action as appropriate.

ActiveStop will not pause the system if there is no mechanical action of the arms. If the front door is not closed by the time the next action starts, the system will pause at that point and arms will not move. This means, you can open the door during an incubation step and if you close it before the next step, the system will simply continue without interrupting the run.

Door locks are optional and can be configured (or removed from the configuration) by a Tecan Field Service Engineer. Door sensors are always active.

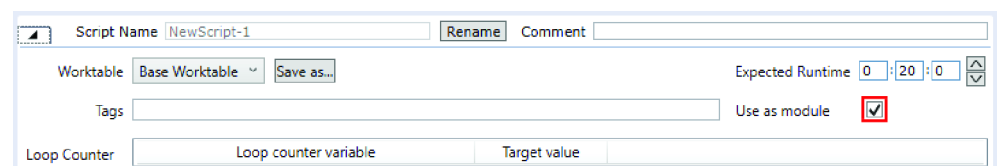
5.8 Modules

5.8.1 General

Module in all its creation behavior is the same as the script. The only difference between module and script is that a module can be inserted in the script and will be treated there as a synchronous subroutine. To run a Module asynchronously, use it in a Subroutine command.

5.8.2 How to create a Module

Module creation is the same as the creation of the script. It can be created from the Scripts or Modules section of control bar menu. To designate a script as a module, check the checkmark "Use as module" and save the script. If the module is created from the Modules section of control bar menu, this checkmark will be set by default.



The screenshot shows a dialog box for creating a new script or module. The 'Script Name' field is set to 'NewScript-1'. There are buttons for 'Rename' and 'Comment'. The 'Worktable' dropdown is set to 'Base Worktable', and there is a 'Save as...' button. The 'Expected Runtime' is set to 0:20:0. The 'Tags' field is empty. The 'Use as module' checkbox is checked. The 'Loop Counter' section has a 'Loop counter variable' field and a 'Target value' field.

Fig. 23: Creating a module

5.8.3 How to use a Module

A Module can be dragged from the Controlbar and dropped into the script editor where it will appear as a Module programming statement. It will automatically contain the name of the called module and the list of parameter variables including their default values that should be transferred into the module. A tooltip function provides the query text information of each parameter variable, if that has been defined, when hovering over the variable. See section 11.11.36 Module for more details on the Module Command.

5.8.4 Other Operations with Modules

Modules can be turned in scripts if the **Use as module** checkmark is unchecked, and the Module is saved. Note: If the module was used in the script and then became a script, the caller script will have a context check error.

5.9 Moving the Arms (Task of Path Finder)

Path Finder controls the movements of all arms between safe points of vectors and travel positions of labware. If necessary arm movements can be influenced by bounding boxes on the worktable, speed parameters in **Labware** attributes or **MicroScript** commands, defining longer **Vectors** and using **Move Arm** commands.

Task of Path Finder

The task of the Path Finder is to move the arms to their target locations and also move other arms out of the way if necessary. Path Finder will avoid all bounding boxes, too. Bounding boxes are the theoretical “no fly zone” around worktable objects and are defined within vControl. Bounding boxes are also what glows when the cursor hovers over an object in the **Worktable** editor. Path Finder will do its best to move arms without issues and calculate problems before they arise. But there are the following limits.

Cannot start from colliding position or cannot move to a colliding position

"Cannot start from colliding position or cannot move to a colliding position" is a Path Finder error message. It does not mean that the arm has physically collided with something. A colliding position means that Path Finder believes that arm is inside of a bounding box of another object at the start of a move or would have to enter a bounding box at the end of the move. The message may look complex, but it is giving hints about what objects are causing the issue (e.g., the left Gripper Finger has an issue and is “colliding” with a hotel). If the colliding position is the current arm position at the start of the movement, then manually re-position the arm to free space and press retry may help.

Limits of Path Finder

For the Robotic Gripper Arm, Path Finder will move the arm from its current position to the Safe Point of the target vector. Movements from the Safe Point of the vector to the End of the vector are not under control of Path Finder, this enables vectors to enter the bounding boxes of the target and surrounding labware or carriers.

For the pipetting arms, the Flexible Channel and Multiple Channel arms, Path Finder will move the arm from current position to the Z travel of the target position. If the arm is inside of a bounding box before the move, Path Finder is allowed to lift the arm in Z only to exit the box. From Z Travel through the pipetting step, the arm is under control of the Pipetting engine and will execute what is defined in the liquid class. Path Finder is not in control of the movement during the pipetting step.

During configuration of a Veya, please verify that all desired movements are physically possible. Path Finder will do its best to move arms without issues and calculate problems before they arise. But there is no software feature that can make a Flexible Channel Arm with a mounted 10 µl Tip reach the bottom of a 100 ml Trough before the ILID block would touch the top.

The pipetting instrument coordinate system

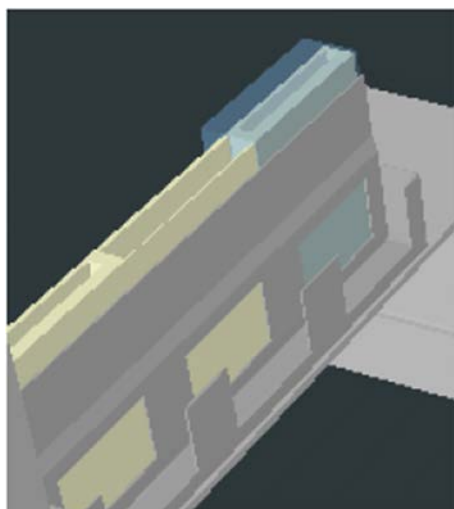
For the instrument coordinate system, the values are relative to the chassis. In the RGA move command, they are relative to that arm. In general the following description applies to all arms, when facing the instrument:

- The **X,Y,Z** “0 Point” or origin is the Left, Front, Worktable level
- **X**: Negative / lower values move to left; Positive / higher values to the right
- **Y**: Negative / lower values move to the front; Positive / higher values to the back
- **Z**: Negative / lower values move lower; Positive / higher move higher
- **Yspread** (FCA Only): Negative / lower values move closer together; Positive / higher values spread apart

- **G** (RGA Only): Negative / lower values move closer together; Positive / higher values spread apart
- **R** (RGA Only): Negative / lower values move clockwise; Positive / higher values move counter-clockwise (viewed from top)
- The coordinate system extends in the negative direction beyond the origin (e.g., RGA with extend Z can reach below the worktable in Z, or negative)

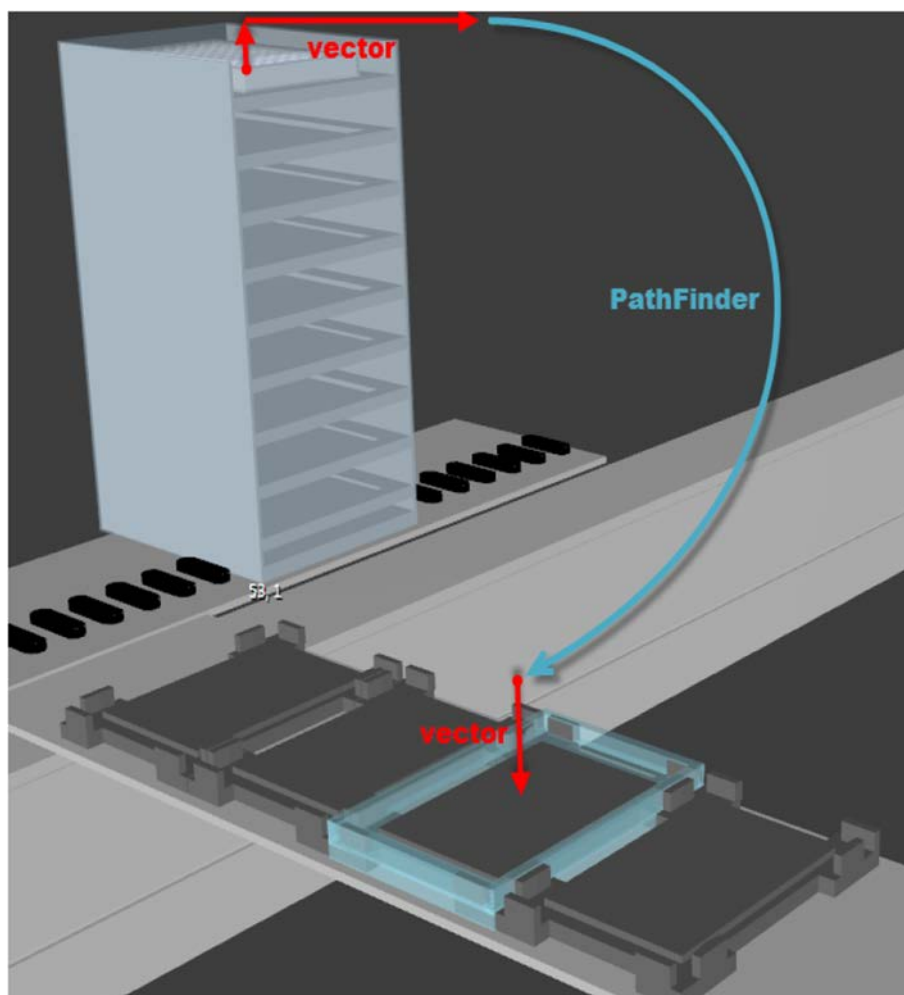
5.10 PathFinder and Robot Vectors

Every object, static objects like the frame of the instrument, the worktable, carriers on the worktable, labware on carriers as well as moving objects like the arms, the mounted tips, have a bounding box definition which defines the outer dimensions. PathFinder calculates a way for the arm to move from a starting point to an end point making sure there is no collision; by ensuring that no bounding box of the moving parts interferes with any other bounding boxes. If movement in a direct line is not possible, Pathfinder will move tips, grippers up or even guide them around objects if necessary. Pathfinder will also move other arms out of the way and ensure they don't collide with any object while moving. This is called an evasion path and if the pathfinder can't find any way to move the arm it will stop and issue a warning to the user. Consider the bounding box as a “no fly zone”.



RGA

For the RGA the Pathfinder calculates the route from the current position to the safe waypoint of the next target vector. However, movements from the Safe Position of a vector to the End Position of the vector are not controlled by Pathfinder. The RGA executes the defined vector from safe to end. While the RGA movement is controlled by a vector the arm is able to move inside bounding boxes. This means potential collisions are not controlled and prevented by Pathfinder while the RGA move is controlled with a vector.



Hence for a plate transfer, the movement is divided into a vector controlled part to fetch the microplate and move it to the Safe position. At the Safe position the bounding box of RGA including its gripped plate must be completely outside of the bounding box of the carrier. At this point the PathFinder takes over control of the movement until the Safe Position of the target carrier. Here again bounding boxes of RGA and plate are not allowed to violate the bounding box of the carrier as long the movement is under PathFinder control. Otherwise PathFinder errors such as for example “The arm is unable to move to the specified end position” are thrown. The last part of the movement from Safe to End position on the target carrier is controlled by the corresponding vector.

Pipetting Arms

For the pipetting arms the Pathfinder calculates the route from Z-travel of the current Labware to the Z-travel of the destination Labware. Once at destination at Z-travel, the arm moves along the Z-axis as defined in the Liquid Class being executed, with Pathfinder not active. This means potential collisions are not controlled and prevented by the Pathfinder.

For example the collision of the Tip adapter with the tube rim, if the tube length is larger than DiTi length.

During configuration of a Veya, please verify all desired movements are physically possible. PathFinder will do its best to move arms without issues and calculate problems before they arise. But there is no software feature that can make a Flexible Channel Arm with a mounted 10µl Tip reach the bottom of a 100ml Trough before the ILID block would touch the top.

Cannot start from colliding position or cannot move to a colliding position. This is a PathFinder error message. It does not mean the arm physically collided. A colliding position means that PathFinder believes the arm is inside of a bounding box of another object at the start of a move or would have to enter a bounding box at the end of the move. The message is giving hints about what objects are causing the issue (e.g. The Left Gripper Finger has an issue and is “colliding” with a hotel).

Robot Vectors

Robot vectors define how to approach a site of a Carrier within the bounding box. Changing these is only needed for custom Carriers and for some non-grid based devices and for teaching purposes : Non-custom carriers have predefined vectors in the software.



Hard crashes can misalign an arm. If after a hard crash you are no longer able to pipette or transport plates: DO NOT attempt to teach the carrier/labware (e.g. 7mm Nest). If "everything" worked until a crash the appropriate service actions must be run in the service software to re-align the arm.

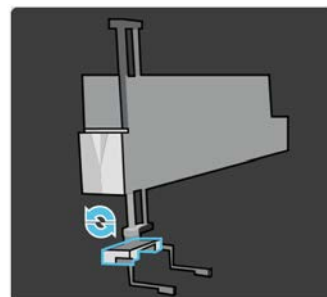
A vector consists of several waypoints. The button Transfer is used to store the current (physical) position of the arm as a waypoint. The three move buttons are used to move to one of the waypoints of the vector.

To create a custom vector click on the button next to the vector dropdown list, define the coordinates for each waypoint and save it. The new vector will then be available to select in the Execute Single Vector Command. Note, that custom vectors are always associated to the carriers where they were created.

Vectors are based on the plate position, not the arm position. This supports different finger types with no need to define a new vector.

Define Coordinates

X [mm] Abs	Y [mm] Abs	Z [mm] Abs	Rotation Z [°] Abs
511.0	270.6	215.4	90.0
358.5	118.2	215.4	180.0



The above two points only rotate the gripper head if eccentric fingers are mounted. Physically these two coordinates are the same in X, Y and Z. The gripper would rotate 90° around Z.

The easiest way to program these coordinates is to put the arm in Zero-G, gently move the arm by hand to the final position, and use the transfer button. vControl will do all the calculations necessary. To teach a position while holding a plate, use the “Fetch Plate for Teaching” direct command before entering Zero-G mode.

Site definitions

A site is a defined space for labware. Properties of the sites are defined in the site definition window, including number of sites per Carrier and their sizes as well as which kind of vector is used to approach the sites (N, NR, W, WR). All grip modes are pre-defined and allowed on each site. The default Transfer Labware command supports all grip modes. The priority however is given to the narrow vector. If you want to change this, only activate the grip mode you want to be used.

Site Definitions

Position Layout: XS 1, YS 1, ZS 1

All dimensions in the table below are specified in [mm].
The rotation is specified in [°].

Grip Modes: N = Narrow, NR = Narrow Reverse, W = Wide, WR = Wide Reverse
Select robot: RGA 1

Site	X-Offset	Y-Offset	Z-Offset	X-Size	Y-Size	Z-Size	Rotation	Location	N	NR	W	WR
1	4.50	5.04	7.00	127.76	85.47	6.00	0.0	Nest7mm_Pos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

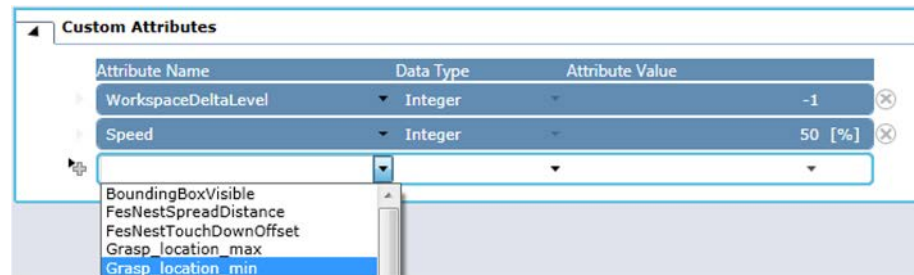
Site coordinates are relative to the carrier.

Allowed Labware types and allowed locations

These two sections contain information on which Labware is allowed on the Carrier, and on which location the Carrier is allowed.

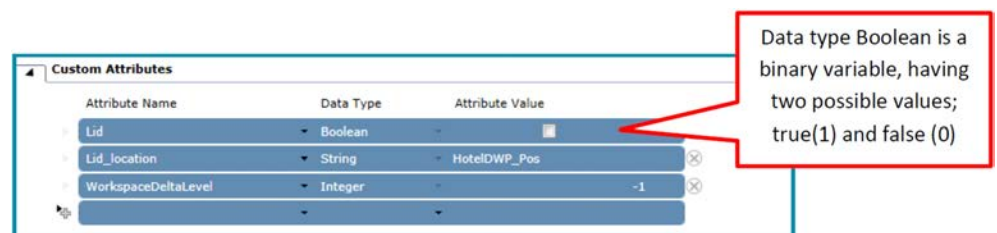
Custom attributes

This section contains attributes, which are settings to define characteristics of Carriers (like is it a waste location), or Carrier-specific settings (e.g. speed for RGA movements from/to this carrier, whether to automatically remove a lid when the microplate arrives at this Carrier, and where to store it and many more).



5.11 Lid Handling

Lid handling refers to the actions of de-lidding and re-lidding a plate or labware. The software will do this automatically if settings are set correctly in the respective labware definitions. 1) The software must know that the labware can have a lid, which doesn't mean it must always have a lid. This is a setting in the group "Robot specific" in the labware editor. 2) The positions where the plate shall have NO lid or must have a lid need to be defined. This is done by activating the attribute Lid and Lid_location in the carrier editor under Custom Attributes. If the attribute Lid is set to false (i.e., unchecked) it means whenever a plate with a lid arrives on this carrier, the lid will be removed. Lid_location defines where the lid is stored. This is done by activating the attribute **Lid** and **Lid_location** in the carrier editor under **Custom Attributes**. If the attribute Lid is set to false (i.e., unchecked) it means whenever a plate with a lid arrives on this carrier, the lid will be removed. **Lid_location** defines where the lid is stored.



Also a lid can be transferred manually with the transfer Labware command outside carrier attributes. In order to allow this the Labware attribute LidHandlingManual needs to set to "True".

How to Change Labware/Carrier attributes on the fly

vControl allows you to change Labware/Carrier attributes on the fly. The following example shows how to switch from automatic to manual lid handling within the script. Refer to [“Functions and Operators” \[306 \]](#) for details about vControl functions.

The screenshot displays a vControl script with the following steps:

- Set Variable:** Name: `lid`, Value: `SetAttribute("liddedplate[001]", "LidHandlingManual", 1)`. A callout box explains: "Lid Handling Manual set to 1, indicates the user will select where to move the lid, enabling the lid to be chosen as the Labware item to move."
- Transfer Labware:** RGA 1: Transfer 'liddedplate[001]' to 'Nest61mm_Pos' position '13'. Target location: `liddedplate[001]`, Target Position: `13`. ☒ Only use selected site.
- Transfer Labware:** RGA 1: Transfer 'liddedplate_Lid[001]' to 'Nest61mm_Pos' position '14'. Target location: `liddedplate_Lid[001]`, Target Position: `14`. ☒ Only use selected site.
- Transfer Labware:** RGA 1: Transfer 'liddedplate_Lid[001]' to '<BASE>'. Target location: `<BASE>`. ☐ Only use selected site.
- Transfer Labware:** RGA 1: Transfer 'liddedplate[001]' to '<BASE>'. Target location: `<BASE>`. ☐ Only use selected site.
- Set Variable:** Name: `lid`, Value: `SetAttribute("liddedplate[001]", "LidHandlingManual", 0)`. A callout box explains: "Lid Handling Manual set to 0, indicates FluentControl will move the lid according to the Labware attributes."

Don't forget to use another **Set Variable** command once manual lid handling is no longer required and you would like vControl to take over and move the lids according to the Labware attributes.

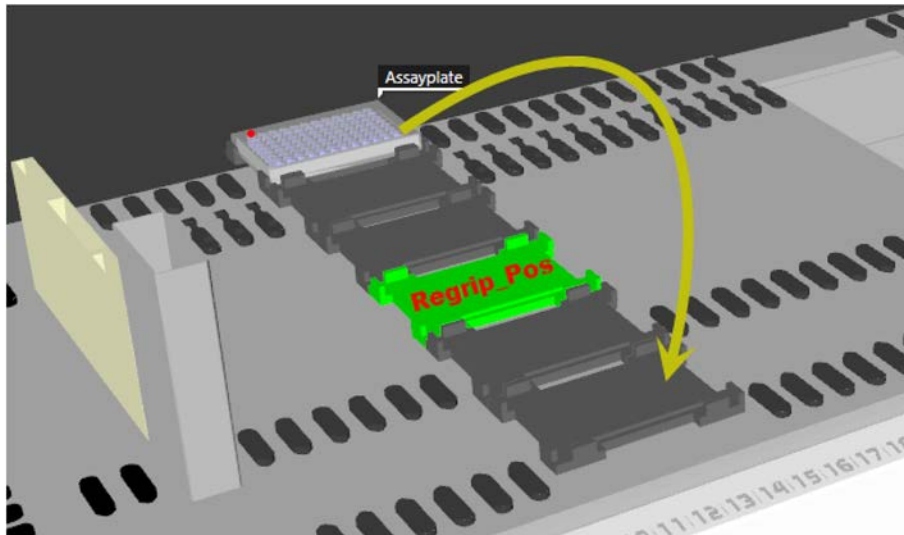
5.12 Regrip

A Regrip station will automatically be used if the source and destination location do not share an allowed vector.

PathFinder does not have any impact on the determination to use a Regrip station. First the RGA driver reads the carrier definitions, figures out which vector to use and whether it has to be offset (e.g., DWP) and then passes the move to PathFinder.

Example

Move a plate from site 1 to 6 and maintain plate orientation



The 6x 7mm nests sit on a deck segment, the 7mm nest allows all vector modes, whereas the deck segment allows different vectors for different sites, based on what is physically accessible. The **Transfer Labware** command picks which vector to use exclusively from the combination of all carriers' allowed vectors under the labware.

7mm Nest

Grip Modes: N = Narrow, NR = Narrow Reverse, W = Wide, WR = Wide Reverse

Site	N	NR	W	WR
1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

6 Landscape 7mm Nest

Site	N	NR	W	WR
1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
7	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Site 1 can only be reached in Wide, whereas the Wide orientation is not allowed on Site 6. To move from Site 1 to Site 6 a Regrip is needed.

There are pre-defined Regrip nests in the vControl database, the "7mm Regrip Nest" and the "61mm Regrip Nest", They are physically the same as the 7mm Nest and 61mm Nest, respectively, but are defined as a Regrip location (Location "Regrip_Pos" defined in the Site Definitions section of the carrier editing tab). Requisite for a Regrip station is to allow all grip orientations.

Site	X-Offset	Y-Offset	Z-Offset	X-Size	Y-Size	Z-Size	Rotation	Location	N	NR	W	WR
1	4.50	5.04	7.00	127.76	85.47	6.00	0	Regrip_Pos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

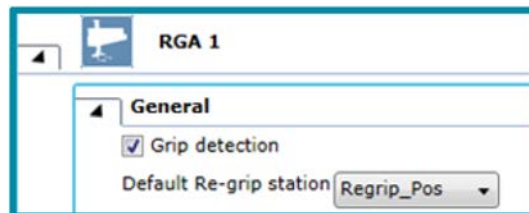
The Transfer Labware command will only use a combination of the vectors:

- Narrow
- Narrow Reverse
- Wide
- Wide Reverse

The order listed above is the order of preference that will be used if more than one combination is feasible.

The grip determination does not check if a vector is defined. It also does not check if it is physically reachable. Undefined Vector & “Colliding Position” are fatal run time errors

Custom Regrip location names can be configured in the RGA driver configuration tab. The name in the RGA configuration must be a location name on the worktable. By default the RGA configuration matches the Regrip Nests.



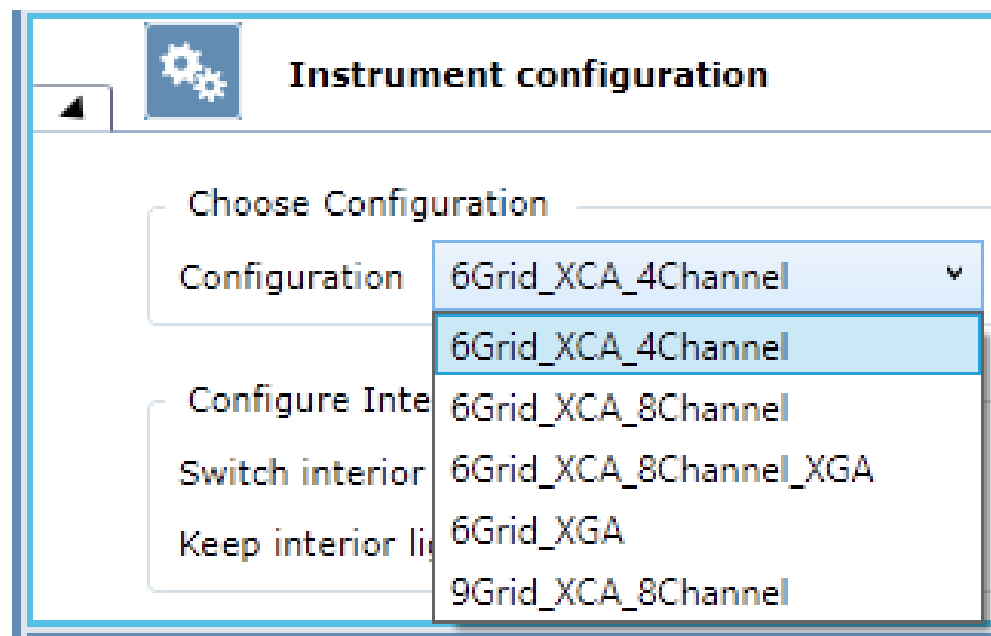
5.13 Simulation Mode

Prerequisite:

- 3DSim is installed (can be found in the installer)

The simulation mode allows to use the application with a simulated instrument, instead of having a real instrument connected to the vControl software.

The installation provides the following instrument configurations for simulation:



1. The configuration can be changed using the menu **Settings -> Configure System**
2. Open Instrument configuration section and select the instrument configuration you want to use in simulation mode.
3. After applying the changes, the software needs to be restarted in order to load the chosen configuration.

There is also a possibility to switch between the real or a simulated instrument. For this go to menu **Run -> Switch to Real Mode OR Simulation Mode**. The software has to be restarted subsequently.

Whenever vControl goes into simulation mode, the selected instrument configuration (see picture above) that is selected in the drop-down list will be applied.

Please note:

If a real instrument is connected and the 3DSim application is running, the vControl application will start in simulation mode.

5.13.1 Instrument Configuration in Simulation Mode

NOTICE

Risk of the script or process aborting.

Risk of loosing connection to the PC After a certain time of inactivity from the user, the PC goes into a sleep mode and the connection to the instrument may be lost. Adjust the PC settings to avoid from going into the sleep mode.

- The PC must be configured as to prevent it from going into **Sleep Mode**.
- The USB power management in Windows must be switched off to avoid losing the connection to the pipetting instrument.

1. StartvControl.

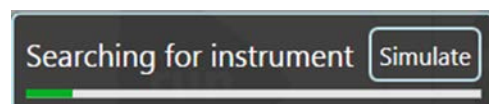


Fig. 24: Searching for instrument

2. Click **Simulate** when prompted.
3. To select your simulated configuration, on the **Settings** menu, click **Configure System**.
4. Open on **General Settings**, the slide **Instrument configuration**.

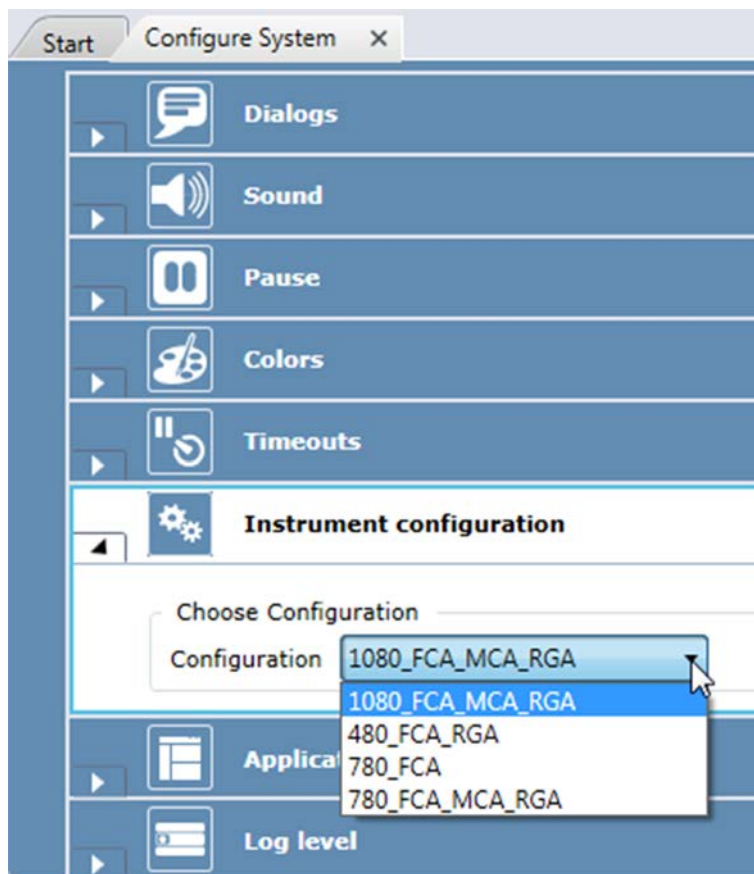


Fig. 25: Configure system

5. From the Configuration list, select an instrument configuration.
6. Click **Save & Close** at the bottom of the window.

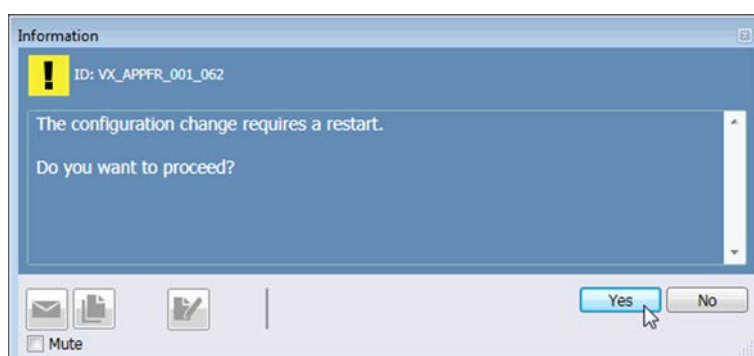


Fig. 26: Configuration change requires restart

7. Click **Yes** to restart, and wait until the button **Simulate** is displayed again.

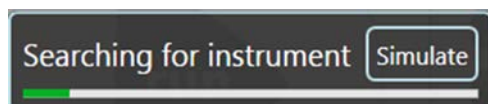


Fig. 27: Searching for instrument

8. Click **Simulate**. It opens the select instrument configuration in simulation mode.

6 User Administration

To activate user management in vControl select **Settings > User Administration...** and click **Activate** in the **User Administration** tab.

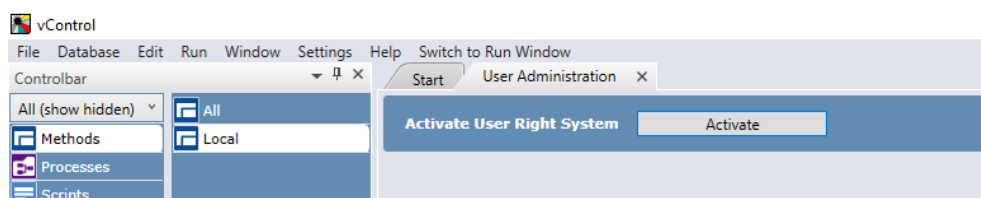


Fig. 28: Activate



You can only change user settings if you have been given vControl administrator access rights.

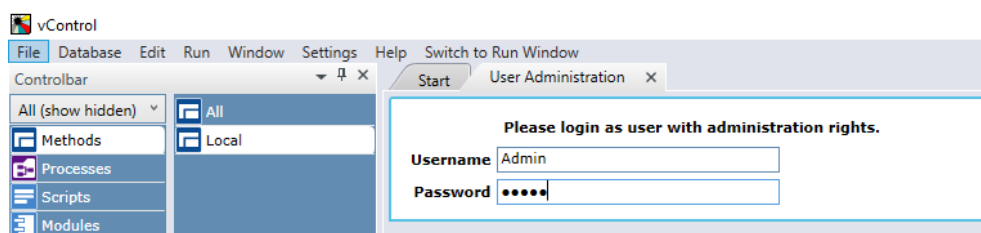


Fig. 29: Log in as Administrator

In User Administration you can change the settings for log in, passwords, notifications, user roles, method permissions, and access rights. You can also activate the use of electronic signatures.

User Administration does not rely on the MS Windows access rights and log in mechanisms which have been configured for your PC and/or for your local network. The Tecan User Management system is a central software component which handles user log in and user authorization for all Tecan software components. It also provides facilities to log out users after a specified period of inactivity and to send warnings by e-mail to a specified person (usually the vControl administrator) if an attempt is made to log in to the Tecan software using an invalid user name and password combination. Furthermore, users are disabled automatically after entering the password incorrectly a specified number of times consecutively.

The Tecan User Management system thus provides an additional security barrier against unauthorized use of the Tecan software. vControl writes log files and audit trails and can create data files. It is important to correctly configure the MS Windows access rights to the respective directories on your PC and/or local network to prevent unauthorized access to and tampering with this data.

6.1 Main Features

- All changes relating to user management require vControl administrator access rights.
- All changes relating to user management are recorded in the user management audit trail.

- User accounts are automatically disabled following a configurable number of consecutive failed attempts to enter the correct user name and password combination. The disabled account can only be reactivated by the vControl administrator.
- vControl does not allow more than one person to use the software simultaneously.
- Users can log out without leaving vControl or interrupting a running pipetting script or process. The same or a different user can then log in to vControl while the script or process is still running.
- Users are automatically logged out after a specified time during which no keyboard or mouse-click activities have been detected.
- User passwords expire after a configurable number of days. The user must then choose another password.
- Passwords must contain a specified minimum number of characters.
- Passwords can be forced to contain numeric digits.
- User names are unique within the system and cannot be assigned more than once.
- The vControl administrator provides a temporary password when setting up a user. When the new user first logs on to the system, he is forced to change the password immediately. This ensures that no one other than the individual user knows the particular user name and password combination.
- If a user forgets his password, the vControl administrator does not have access to the user's password. The vControl administrator is only able to reset the password and provide a new temporary password.
- If the vControl administrator forgets his password, no one has access to the password.
- A specified person (typically the vControl administrator) can be automatically notified by e-mail in case of potential security threats (e.g., a user account has been disabled after a configurable number of consecutive failed login attempts).

6.1.1 Audit Trail

Are you in a regulated industry where audit trails matter?

All changes relating to user management are recorded in the user management audit trail. A log file is written to the following directory:

%PROGRAMDATA%\Tecan\Tecan User Management\v1.1\Audit

The file name is YYYY-MM-DD HH-MM-SS, where YYYY-MM-DD is the file creation date and HH-MM-SS is the file creation time.

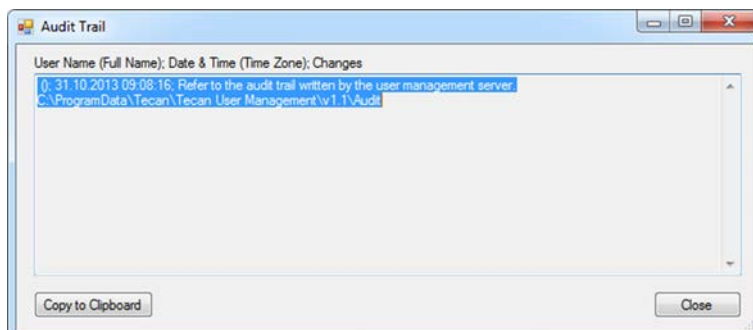


Fig. 30: Audit trail

6.2 Login Settings

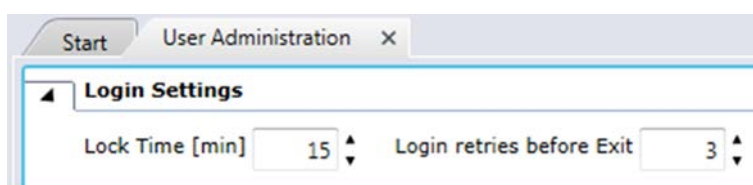


Fig. 31: Login Settings

Lock Time [min]

Users are automatically logged out of vControl after a period of inactivity. Specify that period of time. Logged out users must log in to vControl again to regain control of the program (range: 2 to 1440).

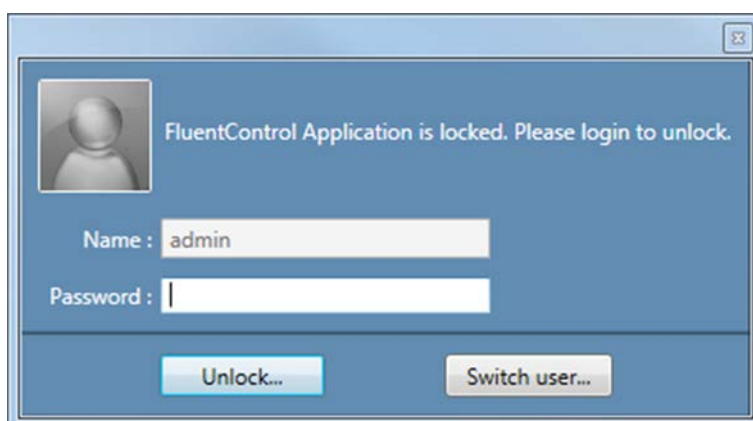


Fig. 32: Log in

Login retries before Exit

User accounts are automatically disabled after a configurable number of consecutive failed attempts to enter the correct user name and password combination. Specify the maximum number of unsuccessful logins (range: 1 to 10).

6.3 Password Settings



Fig. 33: Password Settings

Allow empty passwords. (Except for Administrators)

Select this checkbox to give users—except administrators—access to vControl without entering a password.

Minimum password length

Passwords can be forced to contain a specified minimum number of characters. Specify the required number of characters.

Minimum numeric digits

Specify if passwords should contain one or more numeric characters.

Minimum non-alphanumeric characters

Specify if passwords should contain one or more non-alphanumeric characters.

Minimum number of unique last passwords

Specify the number of previously used passwords that are not allowed to be reused.

Password expiring [days]

User passwords expire after a configurable number of days; the user must then choose another password. Specify the number of days.

6.4 Notify Settings




Fig. 34: Notify Settings

A specified person (typically the vControl administrator) can be automatically warned by e-mail in case of potential security threats (i.e., a user account has been disabled after a configurable number of consecutive failed login attempts). This panel is used to configure the settings for the e-mail warnings.

Notify via Mail

Check this checkbox if you want vControl to send security warnings by e-mail. If you disable this feature, this does not delete the e-mail settings and you can re-enable them later if required.

Mail to

Specify the e-mail address of the recipient (e.g., the system administrator).

SMTP - Server

Specify the address of the SMTP server you want to use for sending the e-mails. You should normally specify the SMTP server address that is recommended by your e-mail provider or e-mail administrator.

Port

Specify the SMTP port number.

Security Protocol

You can choose between **None**, **SSL**, and **TLS**.

Return Address

If you expect to receive replies to the warning e-mails, make sure that you configure an e-mail program on another PC to receive the replies. vControl does not provide facilities to receive e-mails.

Authentication

Check this checkbox if the SMTP server (mail server) requires authentication to allow e-mails to be sent. This is normally the case.

Username

Specify the user name for SMTP authentication.

Password

Specify the password for SMTP authentication.

6.5 User – Role permission

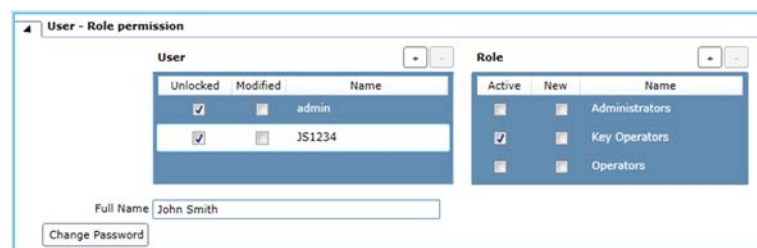


Fig. 35: User - Role permission

The **User** panel lists all users who are currently known to vControl. The **Full Name** of the currently selected user is displayed underneath the panel.

Uncheck the **Unlocked** checkbox to disable the log in for the selected user.



*Never uncheck all of the **Unlocked** checkboxes because you will not be able to log in to vControl again. Do not disable the vControl administrator login.*

The **Role** panel lists all available user groups and shows the groups to which the currently selected user belongs. Check or uncheck the **Active** checkboxes to change the group assignments as required. Users inherit the rights of the groups to which they belong.

Change Password

To change the password of a user click **Change Password** and enter and confirm a new password.

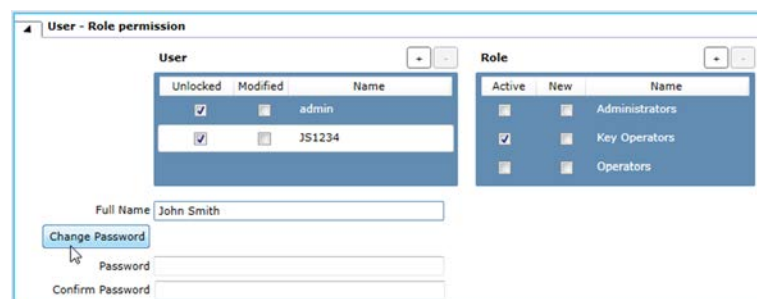
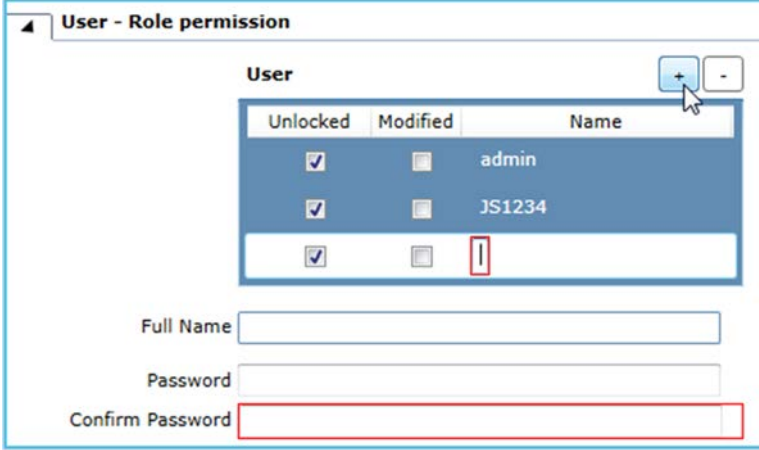


Fig. 36: Changing a user password

Adding a New User



User - Role permission

User

Unlocked	Modified	Name
<input checked="" type="checkbox"/>	<input type="checkbox"/>	admin
<input checked="" type="checkbox"/>	<input type="checkbox"/>	JS1234
<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Full Name

Password

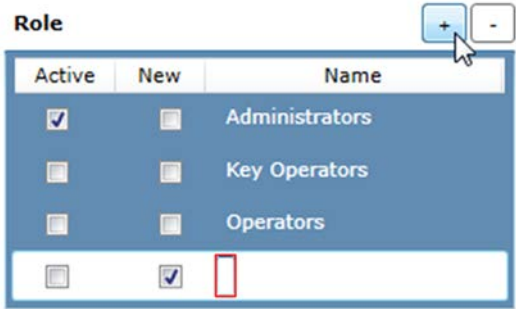
Confirm Password

Fig. 37: Adding a new user

To create a new user account, proceed as follows:

1. Click **[+]**.
2. Enter a unique **Name**.
You can neither change this name nor delete the created user account after clicking **Apply**.
3. You can enter the user's **Full Name**.
4. Enter and confirm a new password.
The password has to comply to the rules that have been defined in **Password Settings**. This initial password is valid for three days. During this time the user has to log in and change his password or the account will be disabled automatically.
5. Click **Apply** to save your changes or **[-]** to abort.

Adding a New Role



Role

Active	New	Name
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Administrators
<input type="checkbox"/>	<input type="checkbox"/>	Key Operators
<input type="checkbox"/>	<input type="checkbox"/>	Operators
<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Fig. 38: Adding a new role

To create a new **Role**, proceed as follows:

1. Click **[+]**.
2. Enter a unique **Name**.

3. Click **Apply** to save your changes or [-] to abort.

6.6 Role – Method permission

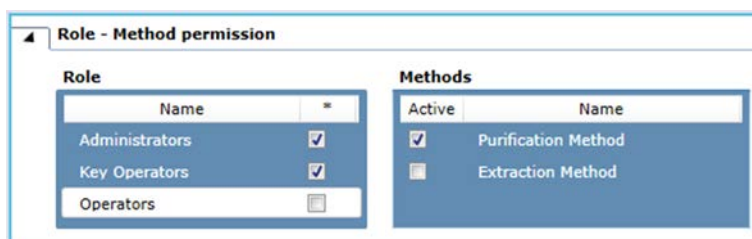


Fig. 39: Method permission

In **Method permission** you can define the methods that are available for a user role. However, what a user can actually do with a method is defined by the access rights of his user role. For more information refer to [“Roles and Access Rights” \[81\]](#).

By checking or unchecking the * checkbox in the **Role** panel you can assign all methods to the selected role or deny access to all methods, respectively.

By checking or unchecking the **Active** checkbox in the **Methods** panel you can assign a specific method to the selected role or deny access to this method, respectively.

6.7 Roles and Access Rights

The **Role - Access Rights** menu lists all available access rights for the currently installed Tecan software components and shows the access rights that are assigned to the selected user role. Users inherit the rights of the role that has been assigned to them.

By default we differentiate between three available Roles:

- **Administrator**
- **Key Operators**
- **Operators**

The system allows to add new roles which can be a new operator, an operator with special rights etc.. The new created role appears in the menu **Roles**. For more information on how to create a new role please refer to [“User – Role permission” \[79\]](#).

You can assign **Access Rights** to the selected **Roles** by checking / unchecking the **Active** box in the menu **Access Rights**.

Access Rights

Role - Access Rights																																																	
Roles	Access Rights																																																
<table> <tr> <th>Name</th><th>*</th></tr> <tr> <td>Administrators</td><td><input checked="" type="checkbox"/></td></tr> <tr> <td>Key Operators</td><td><input type="checkbox"/></td></tr> <tr> <td>Operators</td><td><input type="checkbox"/></td></tr> </table>	Name	*	Administrators	<input checked="" type="checkbox"/>	Key Operators	<input type="checkbox"/>	Operators	<input type="checkbox"/>	<table> <tr> <th>Active</th><th>Name</th></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit user management settings</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit methods</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit scripts</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit liquid classes</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit carriers</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit labware</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit tools</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Run methods</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Run unapproved methods</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Execute direct commands</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Run partial scripts</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Ignore unhandled runtime errors</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Bypass timers</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Import data</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit configurations</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit Touch MethodStarter</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit Touch SystemCare</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Delete/edit worktable</td></tr> <tr><td><input checked="" type="checkbox"/></td><td>Edit processes</td></tr> </table>	Active	Name	<input checked="" type="checkbox"/>	Edit user management settings	<input checked="" type="checkbox"/>	Edit methods	<input checked="" type="checkbox"/>	Edit scripts	<input checked="" type="checkbox"/>	Edit liquid classes	<input checked="" type="checkbox"/>	Edit carriers	<input checked="" type="checkbox"/>	Edit labware	<input checked="" type="checkbox"/>	Edit tools	<input checked="" type="checkbox"/>	Run methods	<input checked="" type="checkbox"/>	Run unapproved methods	<input checked="" type="checkbox"/>	Execute direct commands	<input checked="" type="checkbox"/>	Run partial scripts	<input checked="" type="checkbox"/>	Ignore unhandled runtime errors	<input checked="" type="checkbox"/>	Bypass timers	<input checked="" type="checkbox"/>	Import data	<input checked="" type="checkbox"/>	Edit configurations	<input checked="" type="checkbox"/>	Edit Touch MethodStarter	<input checked="" type="checkbox"/>	Edit Touch SystemCare	<input checked="" type="checkbox"/>	Delete/edit worktable	<input checked="" type="checkbox"/>	Edit processes
Name	*																																																
Administrators	<input checked="" type="checkbox"/>																																																
Key Operators	<input type="checkbox"/>																																																
Operators	<input type="checkbox"/>																																																
Active	Name																																																
<input checked="" type="checkbox"/>	Edit user management settings																																																
<input checked="" type="checkbox"/>	Edit methods																																																
<input checked="" type="checkbox"/>	Edit scripts																																																
<input checked="" type="checkbox"/>	Edit liquid classes																																																
<input checked="" type="checkbox"/>	Edit carriers																																																
<input checked="" type="checkbox"/>	Edit labware																																																
<input checked="" type="checkbox"/>	Edit tools																																																
<input checked="" type="checkbox"/>	Run methods																																																
<input checked="" type="checkbox"/>	Run unapproved methods																																																
<input checked="" type="checkbox"/>	Execute direct commands																																																
<input checked="" type="checkbox"/>	Run partial scripts																																																
<input checked="" type="checkbox"/>	Ignore unhandled runtime errors																																																
<input checked="" type="checkbox"/>	Bypass timers																																																
<input checked="" type="checkbox"/>	Import data																																																
<input checked="" type="checkbox"/>	Edit configurations																																																
<input checked="" type="checkbox"/>	Edit Touch MethodStarter																																																
<input checked="" type="checkbox"/>	Edit Touch SystemCare																																																
<input checked="" type="checkbox"/>	Delete/edit worktable																																																
<input checked="" type="checkbox"/>	Edit processes																																																

Fig. 40: Access Right Overview

Edit user management settings	The User Administration option in the Settings menu will be visible to users with this role. Roles with this access right can activate or deactivate the User Management System and edit any settings in the User Administration tab. For more information see Chapter 6 “ User Administration ” [► 74].
Edit methods	Roles with this access right can create, edit and delete methods. Roles without this access right may only view Methods.
Edit scripts	Roles with this access right can create, edit and delete Scripts and Modules. Roles without this access right may only view Scripts.
Edit liquid classes	Roles with this access right can duplicate, edit and delete Liquid Classes. Liquid classes with a lock symbol cannot be edited or deleted. Roles without this access right may only view Liquid Classes.

Edit carriers	Roles with this access right can duplicate, edit and delete carriers. Carriers with a lock symbol cannot be edited or deleted. Roles without this access right may only view carriers.
Edit labware	Roles with this access right can duplicate, edit and delete labware. Labware with a lock symbol cannot be edited or deleted. Roles without this access right may only view labware.
Edit tools	Roles with this access right can duplicate, edit and delete tools. Tools with a lock symbol cannot be edited or deleted. Roles without this access right may only view tools.
Run methods	Roles with this access right can run methods via the vControl user interface or the touch monitor. A method is only visible in on the touch monitor if the 'Is visible in Touch Tools' box has been checked in the method editor.
Run unapproved methods	This right is required, along with the Edit configurations right, to activate the approval functionality in Configure System. Roles with this access right can approve Methods, Processes, Scripts and Modules. Roles with this access right can run unapproved Methods, Processes, Scripts and Modules. For more information on method approval, see section Method Approval.
Execute direct commands	Roles with this access right can run direct commands. For more information on direct commands, see section Direct Commands.
Run partial scripts	Roles with this access right can run partial scripts.
Ignore unhandled runtime errors	Roles with this access right can ignore runtime errors and continue the run without handling them.
Bypass timers	Roles with this access right can bypass timers.
Import data	Roles with this access right can import data into the vControl database by using the Import function in the Database menu. For more information on importing data, see section “Import Mode” [400] .
Edit configurations	Roles with this access right can view and change any setting in the Configure System option in the Settings menu.
Edit Touch MethodStarter	Legacy Role with no current functionality.
Edit Touch SystemCare	Legacy Role with no current functionality.

Delete/edit worktable	Roles with this access right can create, duplicate edit and delete worktables. Roles without this role may only view worktables.
Edit processes	Roles with this access right can create, edit and delete processes. Roles without this role may only view processes.

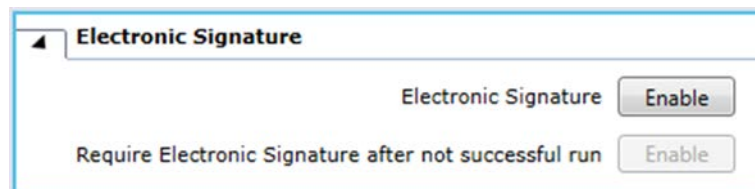
Administrators By default the **Administrators role** has all **Access Rights** active, except Run unapproved methods. Tecan strongly recommends not to change these settings for the **Administrators** role. Tecan also recommends to create at least two users with the **Administrators** role, in case one Administrators user locked out of the software unexpectedly.

Key Operators By default the role **Key Operators** has ALL access rights—except **Edit user management settings**

Operators By default the following access rights are assigned to the role **Operators**:

- **Run methods**
- **Execute direct commands**
- **Edit Touch MethodStarter**
- **Edit Touch SystemCare**

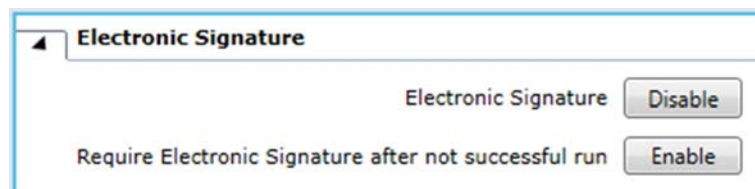
6.8 Electronic Signature



The screenshot shows a window titled "Electronic Signature". It contains two settings: "Electronic Signature" with an "Enable" button, and "Require Electronic Signature after not successful run" with an "Enable" button.

Fig. 41: Enabling electronic signature

Click **Enable** to enable **Electronic Signature**.

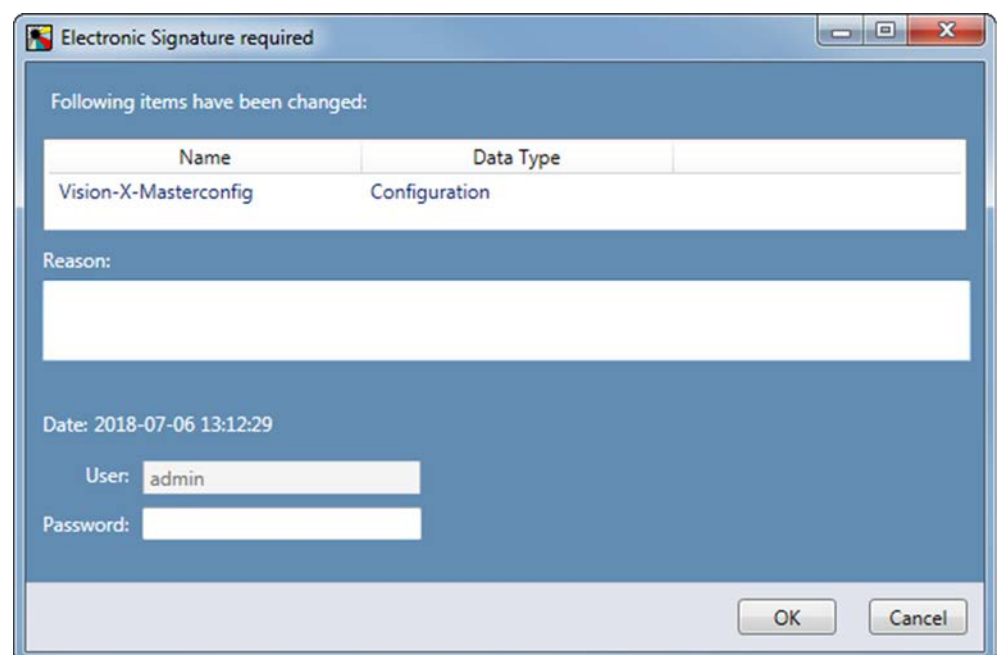


The screenshot shows the same "Electronic Signature" window, but the "Electronic Signature" button is now "Disable", and the "Require Electronic Signature after not successful run" button remains "Enable".

Fig. 42: Disabling electronic signature

If you want the user to sign electronically after each run that has not been successful click **Enable**.

When your electronic signature is required (e.g., for disabling the Electronic Signature) you need to enter a **Reason** and your **Password**.



The screenshot shows a dialog box titled "Electronic Signature required". It contains the following information:

- Following items have been changed:
- Table with 2 columns: Name, Data Type
- Row 1: Vision-X-Masterconfig, Configuration
- Reason: (text input field)
- Date: 2018-07-06 13:12:29
- User: admin (text input field)
- Password: (password input field)
- Buttons: OK, Cancel

Fig. 43: Electronic signature

6.9 Deactivate User Management

To deactivate user administration select **Deactivate User Management** and click **Deactivate**.

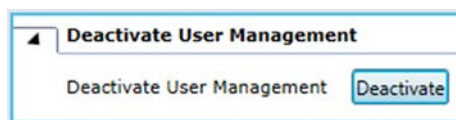


Fig. 44: Deactivating user administration



With user management disabled, it is not necessary to enter a user name or password when vControl is started, and all users will be given vControl administrator access rights automatically. Be aware of the security risk if you deactivate user management and make sure this is permitted by the security procedures of your laboratory or company.

6.10 User Management Configuration Options

6.10.1 Local User Login and Local Roles

This is the standard configuration in vControl when User Administration is activated.

For more information on how to create users and user roles with standard configuration refer to:

- [“User – Role permission” \[► 79\]](#)
- [“Roles and Access Rights” \[► 81\]](#)

To switch back to Local User Login and Local Roles after using Windows Active Domain Login, proceed as follows:

- ✓ Windows administrator rights are needed to run this application.
1. Exit vControl.
 2. Navigate to the directory C:\Program Files (x86)\Tecan\Tecan User Management\v1.1.
 3. Start the **SwitchProviders** tool by double-clicking **Tecan.UserManagement.SwitchProvider.exe**.



The status of the radio buttons shows which of the two user management systems is currently being used by vControl.

4. Select **Encrypted XML Provider**.
5. Click **Set**.

6.10.2 Windows Active Domain Login



The Tecan User Management System can be configured to use the Microsoft Windows user database (with Windows login names) instead of the Tecan user database. This avoids the need to remember two different user names and passwords—one for Microsoft Windows and a different one for vControl.

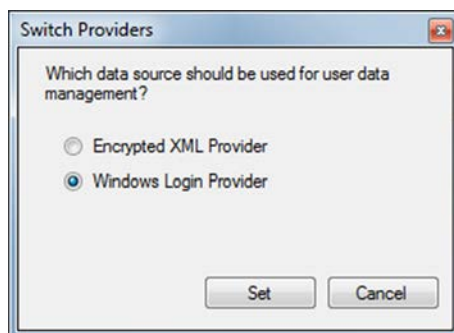
The User Management System is implemented as a server, which runs as a Windows service. The Tecan user management server normally manages users and roles using an encrypted XML database, but it can be configured to use the Microsoft Active Directory user and groups database or the Windows local machine user database instead (in read-only mode). If your company uses Microsoft Active Directory for managing users and groups and you have more than one vControl system, this allows you to manage Tecan users and groups centrally for all of your systems.

The Tecan User Management System requires domain usernames to be 'username@FQDN' where FQDN stands for Fully Qualified Domain Name. Subdomains are not supported.

Use the **SwitchProviders** tool to switch between using the Tecan user management database (encrypted XML file) and using the Windows user management database:

- ✓ Windows administrator rights are needed to run this application.

1. Exit vControl.
2. Navigate to the directory C:\Program Files (x86)\Tecan\Tecan User Management\v1.1.
3. To start the **SwitchProviders** tool, double-click **Tecan.UserManagement.SwitchProvider.exe**.



The status of the radio buttons shows which of the two user management systems is currently being used by vControl.

4. Select **Windows Login Provider**.
5. Click **Set**.



After switching to the Windows user database, user accounts for vControl must be created, edited, and deleted using Microsoft Active Directory or the User Accounts feature in the Windows Control Panel and no longer in vControl.

Tecan User Groups and User Roles

Windows users are assigned to roles (user groups) in vControl by assigning them to specially created Windows groups that use the following naming convention:

Tecan_<vControl_role_name>

(i.e., the string "Tecan_" is prefixed to the vControl role name)

To run vControl using a Windows login, you must be a member of at least one of these Windows groups (depending on the vControl rights you require):

- Tecan_Administrators
- Tecan_Key_Operators
- Tecan_Operators



These roles must be defined by an IT administrator on the domain level (not on the local computer).



Required Active Directory Domain Permissions

For the Tecan User Management service to work properly with the Active Directory domain, all accounts involved require the proper reading permissions in the domain.

The accounts involved with Tecan User Management are:

- *The account(s) that the User Management service run on (by default the local SYSTEM account)*
- *The accounts that shall be able to log into Tecan User Management*
- *The Windows user account must be in the local Windows user group 'Users'*

*These accounts will require reading permissions on all user accounts in the domain that shall be able to log in to Tecan User Management. More specifically, this includes the permissions **Read general information** and **Read group membership**.*

6.10.3 Mixed Configuration: Windows Active Domain Login With Local Roles

This configuration is recommended for large organizations that want to use Active Domain logins but limit the access rights of individual users (i.e., Administrators, Key Operators, Operators) to specific sets of instruments rather than granting them globally for every Tecan instrument of that organization.



The Tecan User Management System can be configured to use the Microsoft Windows user database (with Windows login names) instead of the Tecan user database. This avoids the need to remember two different user names and passwords—one for Microsoft Windows and a different one for vControl.

The Tecan User Management System

The User Management System is implemented as a server, which runs as a Windows service. The Tecan user management server normally manages users and roles using an encrypted XML database, but it can be configured to use the Microsoft Active Directory user and groups database or the Windows local machine

user database instead (in read-only mode). If your company uses Microsoft Active Directory for managing users and you have more than one vControl system, this allows you to manage Tecan users centrally for all of your systems.

When using local roles, login is only possible for users who are assigned to one of the special Tecan groups on the active domain—the local rules only provide permissions but have no effect on the login and password verification.



Tecan User Groups And User Roles

After switching to the Windows user database, user accounts for vControl must be created, edited, and deleted using Microsoft Active Directory or the User Accounts feature in the Windows Control Panel and no longer in vControl.

Windows users are assigned to roles (user groups) in vControl by assigning them to specially created Windows groups that use the following naming convention:

Tecan_<role_name>
(i.e., the string “Tecan_” is prefixed to the vControl role name)

To run vControl using a Windows login, you must be a member of at least one of these Windows groups (depending on the vControl rights you require):

- Tecan_Administrators
- Tecan_Key Operators
- Tecan_Operators

For more information on user role names and rights in vControl refer to [“User – Role permission” \[p. 79\]](#).



Preparations

You need Windows administrator rights for completing the following procedure.

1. Exit vControl.
2. Navigate to the directory C:\Program Files (x86)\Tecan\Tecan User Management\v1.1.
3. Make a copy of the file Tecan.UserManagement.Server.exe.Config and store it for later use.

Procedure

1. Run Tecan.UserManagement.SwitchProvider.exe as an administrator.
2. Switch to the **Encrypted XML Provider** option and click **Set**.



The program will ask to restart the UMS Server; click **Yes**. The program should confirm that the server was restarted after a short period of time.

3. Navigate to the directory `C:\Program Files (x86)\Common Files\Tecan\Tecan User Management\v1.1`
4. Run **Tecan.UserManagement.Manage.exe**.
If no administrator account exists yet in the XML Provider, the program will show a dialog to create one. Log in with the administrator account.
5. Create a new user. As the **User Name**, enter the full name of an Active Directory user (e.g., "user1@domain.com"). It is imperative that the name matches exactly with an existing Active Directory user. The full name and password can be set to any value as they are handled by the Windows Provider later on. Assign the new user to the **Administrators** group. Click on **OK** to create the user and exit the program.
6. Navigate to the directory `C:\Program Files (x86)\Tecan\Tecan User Management\v1.1`.
7. Open the `Tecan.UserManagement.Server.exe.Config` file with an editor program, such as "Notepad".
8. Go to the `<tecan.usermanagement.1.1>` section.
9. Find the row that starts with "`<membership defaultProvider="`" and highlight this row and all following rows until the row "`</roles>`".
10. Replace the highlighted rows with the following content:

```
<membership defaultProvider="WindowsMembershipProvider">
  <providers>
    <remove name="NamedPipeMembershipProvider" />
    <remove name="XmlMembershipProvider" />
    <add roleProvider="XmlRoleProvider"
profileProvider="XmlProfileProvider"
watchUserActivity="False" perThreadSessions="True"
name="XmlMembershipProvider"
type="Tecan.UserManagement.Security.XmlMembershipProvider"
/>
    <remove name="WindowsMembershipProvider" />
    <add roleProvider="XmlRoleProvider"
watchUserActivity="False" perThreadSessions="True"
name="WindowsMembershipProvider"
type="Tecan.UserManagement.Security.WindowsMembershipProvi
der" />
  </providers>
</membership>

<roles defaultProvider="XmlRoleProvider">
  <providers>
    <remove name="NamedPipeRoleProvider"/>
    <remove name="XmlRoleProvider" />
    <add name="XmlRoleProvider"
type="Tecan.UserManagement.Security.XmlRoleProvider"
membershipProvider="WindowsMembershipProvider"/>
  </providers>
</roles>
```

11. Save the changes and close the editor.

12. Restart the User Management Server.



Do not attempt to run `Tecan.UserManagement.SwitchProvider.exe` after editing the `Tecan.UserManagement.Server.exe.Config` file.

In that case, you have to replace the `Tecan.UserManagement.Server.exe.Config` file with your stored copy that has not been edited before running `Tecan.UserManagement.SwitchProvider.exe`.

7 Menus

A menu is a group of the vControl main commands arranged by category, such as **File, Database, Edit, Run, Window, Settings** and **Help**. The menus are displayed on the menu bar located near the top of the main window.

The menus of vControl are only available if you have been given vControl administrator or key operator rights in the Tecan User Management system. Operators can only run existing methods.

For details about access rights and user levels, refer to sections User Qualification and Access Right.



*Most of the features and dialog boxes in vControl are provided with context-sensitive help that can be accessed with the **F1** key.*

7.1 File

Use this menu to create a Method, Script, Process, Module or Worktable with the **New** command and **Open** an existing **Method, ProcessScript, Module, Liquid Class, Worktable, Carrier, Labware**, or **Tool**. Create a printout of the corresponding editor content, i.e., your method or script etc., with the **Print** command.

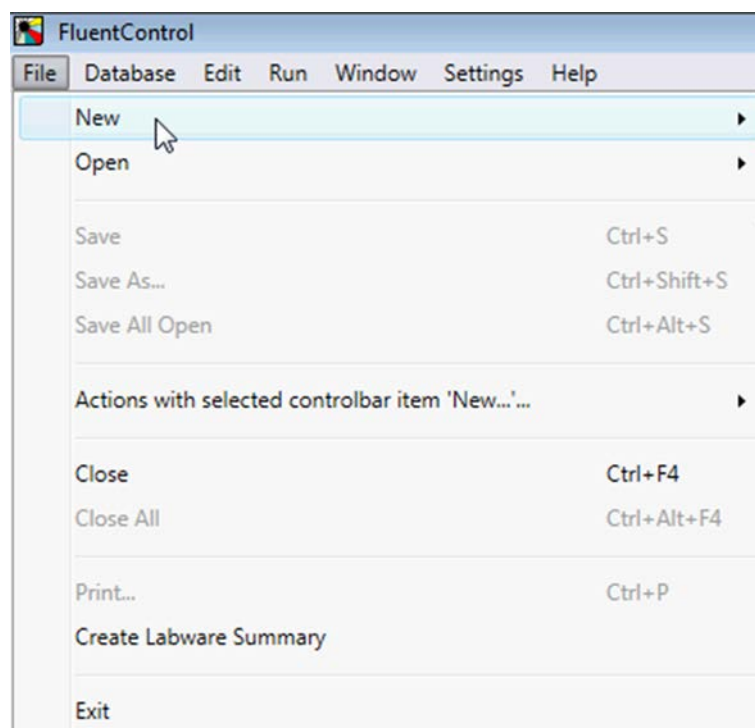


Fig. 45: File Menu

The **File** menu contains the following commands:

- **New:** The command to create a new method, process, script, module or worktable.

- **Open:** The command to edit an existing method, script, process, module, liquid class, worktable, carrier, labware, or tool.
- **Save:** The command saves the current editor content (e.g., a method or a script). Shortcut: Ctrl+S.
- **Save As:** The command allows you to save a copy of the current editor content under a new name. Shortcut: Ctrl+Shift+S.
- **Save All Open:** The command performs the save operation on each editor. Shortcut: Ctrl+Alt+S.
- **Actions with selected Controlbar:** Item to get the shortcut menu of the current object selected in the **Controlbar**.
- **Close:** The command closes the current editor: Shortcut: Ctrl+F4.
- **Close All:** The command closes all editors. Shortcut: Ctrl+Alt+F4.
- **Print:** The command to create a printout of the currently active editor content. If the currently active editor is, for example, a script editor, the script which you are editing will be printed.
- **Create Labware Summary:** The command to export a plain text file with all parts, their part number (Product number), type and quantity present in the currently open Worktable editor.
- **Exit:** The command to close the software. If unsaved data exists, you will be asked if those data objects should be saved before closing the program, or if the exit operation should be cancelled: Shortcut: Alt+F4.

7.2 Database

7.2.1 Overview

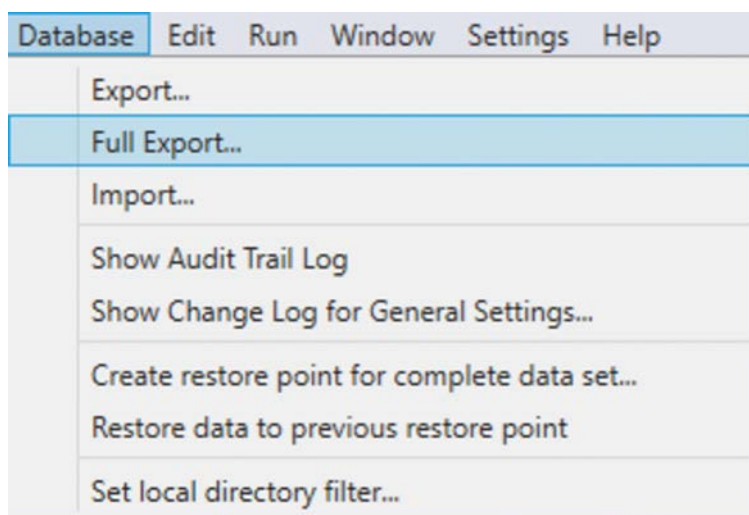


Fig. 46: Database menu

The **Database** menu contains the following commands:

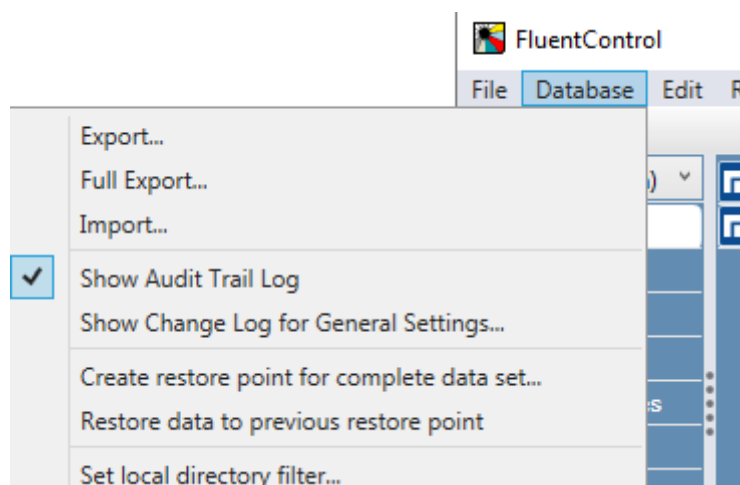
- Use the **Export...** command to save your object, for example, a script to an External Deep Save Data File.
- The **Full Export...** command saves the complete database to a file. Labware and tool templates as well as instances are also part of the exported data set.

- Use the **Import...** command to read a previously exported External Deep Save Data File (a file containing one or more objects) and select the objects to be imported into the database.
- The **Show Audit Trail Log** The Show Audit Trail Log command displays the audit trail log at the Start tab.
- The **Show Change Log** command displays the change log for the selected object (e.g., **General Settings**).
- Create restore point for '<>'... Creates a restore point for the currently opened data object. This can be a method, process, script, module, liquid class, worktable or carrier/labware.
- The **Create restore point for complete data set...** command creates a restore point for the complete data set. Restore points are created automatically with every run for any object involved in the run that has changed. Additionally, restore points can be manually triggered with the **Create restore point for complete data set...** command.
- The **Restore data to previous restore point** the Restore data to previous restore point command first creates a restore point of the database and then leads to a Revision history window, where the user can select a certain point in the history to go back to.
- The **Set local directory filter...** command changes the local directory for the data set.

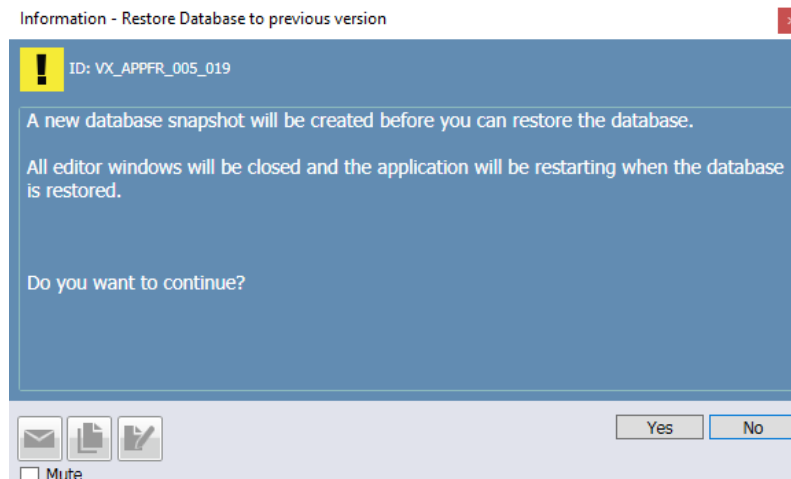
7.2.2 Restore Database

This section explains how to restore a database to a previous version in vControl.

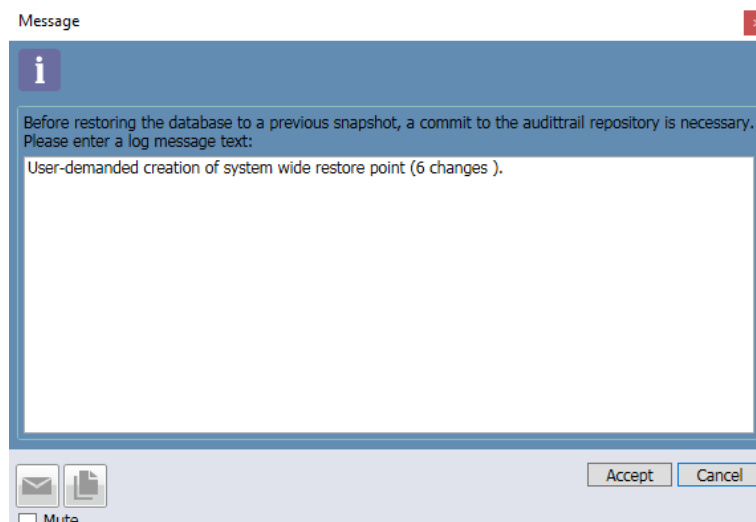
1. Click **Database** in the menu bar
2. Click **Restore data to previous restore point**.



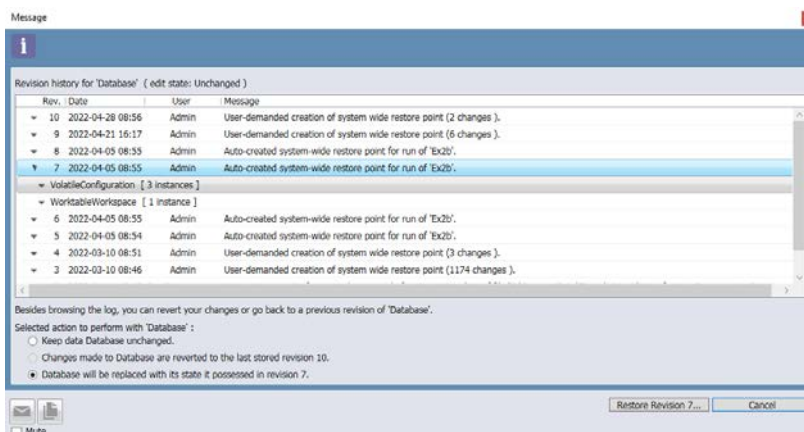
3. Click **Yes** to confirm the next information prompt



4. Edit Restore point's message text if desired, then click **Accept** and then click **OK**.



5. In the next prompt, select the desired restore point from the list.
6. Choose the option **Database will be replaced with its state it possessed in Revision XX**.
7. Click **Restore Revision XX...**, where XX indicates the revision of the chosen restore point.



8. Confirm the next prompt with **Yes, revert**.

vControl restores the database and restarts. After the restart, the database is restored to the selected revision. The most recent 5000 restore points are accessible. If you need to restore to an inaccessible restore point, contact our help desk.

7.2.3 Export

When saving a script, dependent objects (e.g., worktable, carrier, labware, liquid class) and external file references used in script commands (e.g., worklists, VB scripts) will be linked to the script. When exporting a script, dependencies and referenced files are included in the export file.

To export a database object proceed as follows:

1. In the **Database** menu, click **Export...** or **Full Export....**

*The **Export** dialog window opens.*

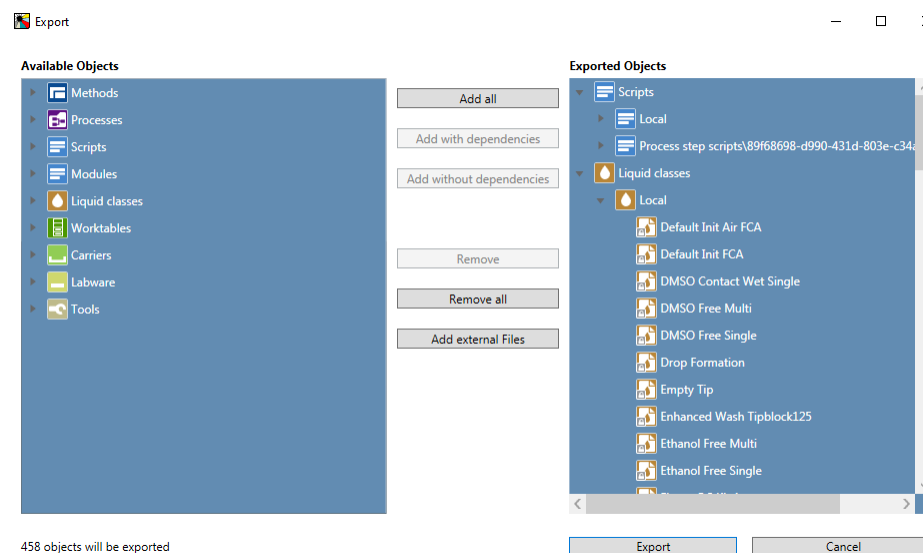
*The **Available Objects** list on the left is expandable and contains items that are available for export.*

The **Exported Objects** list on the right contains items that are currently selected for export (empty in the image below).



2. Select the items to be exported by using one of the following options:
Add all adds all items in the database to the **Exported Objects** list.
Add with dependencies adds all selected items to the **Exported Objects** list—including their referenced scripts, liquid classes, worktables, carriers, labware and tools.
Add without dependencies adds all selected items to the **Exported Objects** list—without their referenced objects. Note that if any of the referenced items is missing, it may not be possible to import these items successfully to the target system.
3. To include external files in the export (e.g., images, PDFs, etc.), click **Add external Files**.
*In the file browser that opens, select the files and click **Open**.*

*The selected external files will be added to the **Exported Objects** list.*



4. Click **Export** to export all items of the **Exported Objects** list.

7.3 Edit

In addition to the **Undo**, **Redo**, **Undo List**, **Redo List**, **Clear undo/redo history**, **Cut**, **Copy**, **Paste**, and **Delete** commands, this menu also provides a search and replace feature for liquid classes and arms.

The **Edit** menu contains the following commands:

- **Undo/Redo**: The Worktable, Script and Liquid Class editors supports undoing or redoing of changes. The Labware and Carrier editors supports undoing or redoing of changes only after any changes have been saved.
- To **Undo/Redo** a change, use Ctrl+Z, Ctrl+Y or Undo/Redo in the Edit menu.
- **Undo List**: Undo list is an expandable menu showing the available actions to undo.
- **Redo List**: Redo List is an expandable menu showing the available actions to redo.
- **Clear undo/redo history**: Use the command to create a clean starting point or to reduce clutter.

7.3.1 Find and Replace Liquid Classes

With the **Find and Replace Liquid Classes** feature you can change the used liquid classes in open scripts. It allows to replace the used liquid classes for selected script commands by another liquid class in batches instead of having to manually change liquid classes.

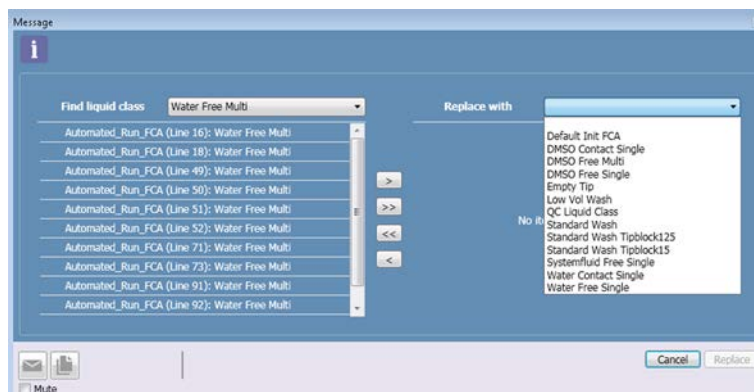


Fig. 47: Search and Replace Liquid Classes dialog

A dialog shows all found script commands in open editors filtered by the currently selected liquid class.

1. To achieve this, in the **Find liquid class** list, click the liquid class you want to replace.
2. in the **Replace with** list, select the new liquid class, which should replace the items found in open editors.
3. If you are satisfied with your selection of commands that you want to get a new liquid class, click **Replace**.
4. Otherwise, click **Cancel**.

7.4 Run



All scripts that are intended to be used in routines not just by the key operator during development, but also by the operator during operation, should be added to a method, so they can be started from the touch monitor.

The **Run** menu contains the following commands:

- **Initialize Instrument:** The command is only available if no run is active or paused. If the command is not available it is grayed out. If the instrument is not or only partially initialized, this command starts the instrument initialization. If the instrument is already initialized, this command shows a dialogue telling the user that the instrument is already initialized and asking them if they want to force an initialization, and offering UI-Controls for **Yes** and **No**.
- **Prepare Method:** The command is only available in edit mode, otherwise grayed out. If the edited object is a script, this command opens a new method editor tab with a method containing the edited script and switches the software to run mode. If the edited object is a method, this command switches the software to run mode.
- Switch to **Simulation Mode:** This option is available when being connected to a physical instrument. Selecting this option will switch the instrument state to simulated and when starting a run no physical actions will be performed. Use this when testing a script to verify correct behaviour. For more information about simulated mode, refer to chapter 5.4
- Switch to **Real Mode:** This option is only available when being in simulated mode. This will again connect to the physical instrument (if available).

- **Switch to Run Mode:** This button will switch the window from edit window to the run window. This is only available if only one screen is available and hence only one window shown.
- **Run:** The command can be selected from the menu for a script and then the user is asked if he wants to use the saved DiTi positions or start with fresh DiTi racks. Additionally, the Run action can be aborted by selecting cancel.
- **Pause/Resume:** Pauses/resumes the current run depending on the state. The command reads **Pause** if a run is being executed and the software is in run mode. If a run is paused, the command reads **Resume**. If no run is being executed, the command is grayed out.
- **Stop:** The command is only available in run mode when a run or initialization is being executed, otherwise it is grayed out. Stops the current run or initialization.
- **Auto Context Check:** A check mark indicates that the context check is active during editing. To switch the context check off, remove the checkmark by clicking on this menu option. If the context check is inactive and a run is started, a dialog will warn the user that the context check is switched off and the run may be interrupted.

7.5 Window

- **Lock:** If User Administration has been activated, use this option to lock vControl. Active runs will not be interrupted. If User Administration is not activated, the option is grayed out.
- **Switch User:** If User Administration has been activated, use this option to switch between users. If User Administration is not activated, the option is grayed out.
- **Reset Layout:** Reset window placement to the default layout.

A checkmark indicates which windows are currently active:

- Start page
- Controlbar
- Worktable
- Gantt chart
- Worktable Orphaned Labware
- Trace view
- Infopad
- MoveTool
- Variable View



NOTICE

Windows can only be closed with the small x on the right top corner of the respective window. Un-check opened windows in the menu is not possible.

7.6 Settings

The Settings category contains the following commands:

- **Controlbar Views**

- **Color Schemes**
- **External links**
- **Tecan Licensing Client**
- **Configure System**
- **User Administration**

7.6.1 Controlbar Views

Click **Settings > Controlbar Views** to change the view according to your preference. There are three supported views for the Controlbar:

- **Expert:** Small icons and list view. Individual window for each level.
- **Power:** List view. Windows slide when different levels are selected.
- **Novice:** Icon view: Windows slide when different levels are selected.
- **Toggle smart mode:** Context relevant icons. This option will allow you to switch between default mode (all items visible) and smart view.

We recommend to start with **Novice**.

7.6.2 Color Schemes

There is currently only one color scheme available.

7.6.3 External Links

This option will open the available links in your default browser. To add additional links to this menu, copy any URL files to

`C:\ProgramData\Tecan\VisionX\ExternalLinks`

The file name will be displayed in the External Links menu.

7.6.4 Tecan Licensing Client

The client requests a license and add your license file to the Licensing Client Module (separate window). Please refer to [“Activate License” \[► 29\]](#).

7.6.5 Configure System

This will open the System Configuration panel:Configuring Your Drivers

7.6.6 Home Position

A home position is defined for each robotic arm. After initialization of the instrument, the robotic arms move to their home positions. The default home position of the FCA module, for example, is over the nearest wash/waste station with Z-axes on Z-travel position. To configure home positions, use the Configure System command on the **Settings** menu, expand the devices section and click **Home-positions**.

7.6.7 User Administration

Please refer to [“User Administration” \[► 74\]](#).

7.6.8 Method Approval



To use Method Approval user management has to be activated. Refer to section [“User Administration” \[74\]](#).

With Method Approval operators can only run methods that have been released by a power user.

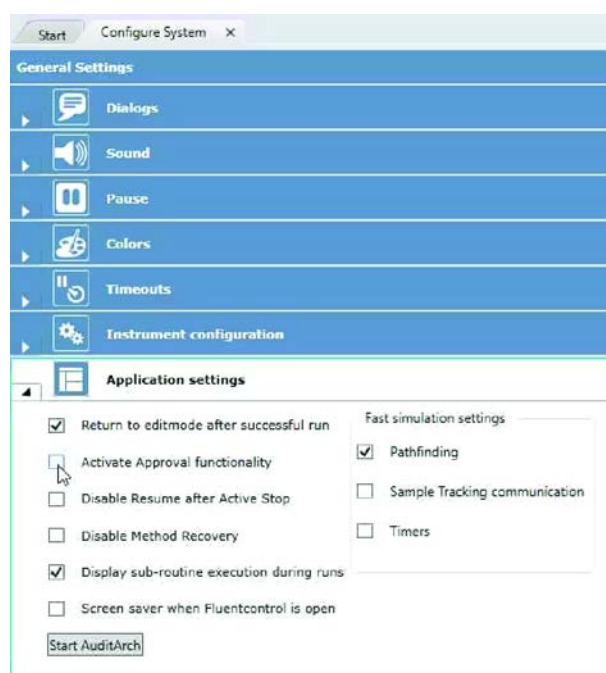
All other methods are flagged:

- Methods that are not released are flagged **red**.
- Methods that have been released and subsequently edited are flagged **yellow**.

The power user can release a method by right-clicking the method and selecting **Release** in the context menu.

To (de-)activate Method Approval proceed as follows:

1. Click **Settings > Configure system...**
2. Click **General Settings > Application settings**.
3. (De-)select **Activate Approval functionality**.



Releasing the first method takes a bit of time, as vControl approves all of the subcomponents of the method—i.e., liquid classes, carriers, labware, worktables, etc.

Subsequent releases for methods that contain the same subcomponents take much less time, as the components have already been released.

If a script, liquid class, labware, etc., is changed, it will cause all methods using it to revert to unreleased status—but re-releasing it will quickly restore the release status of the methods.

7.6.9 Log Level

The default log level in vControl is Information. For more details in log files, the log level can be changed to “Debug”.

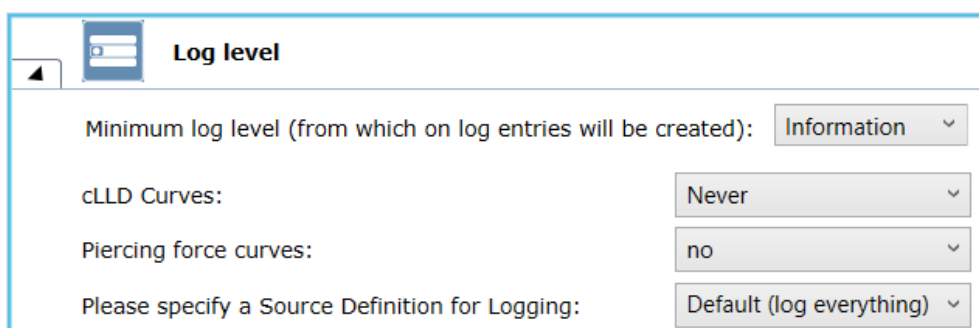
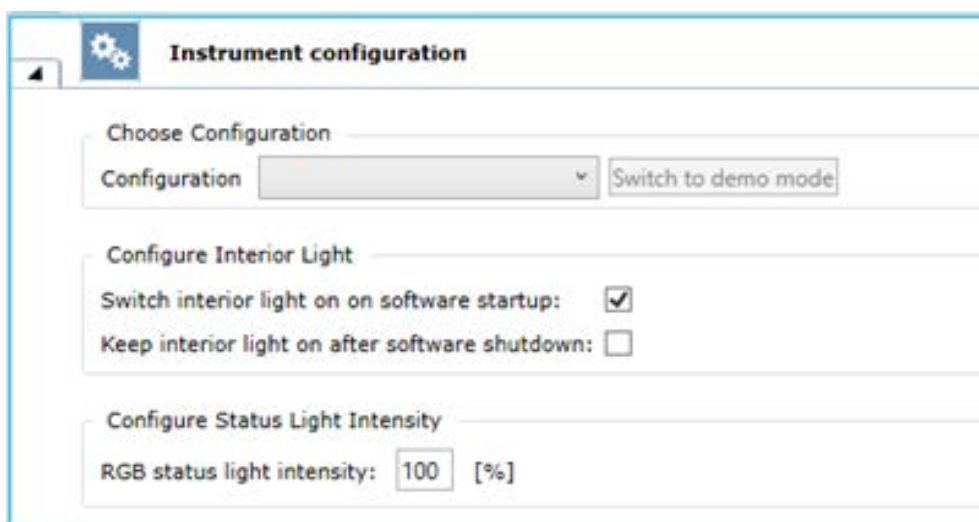


Fig. 48: Log Level

7.6.10 Status Light Intensity

The status light intensity is configurable through “Configure System”. The default value is 100%. Lowering the intensity results in a lower brightness of the status light.



7.7 Help



*Most of the features and dialog boxes in vControl are provided with context-sensitive help that can be accessed with the **F1** key.*

Help can also be accessed by clicking **Open** on the Help menu.

Information about the Software

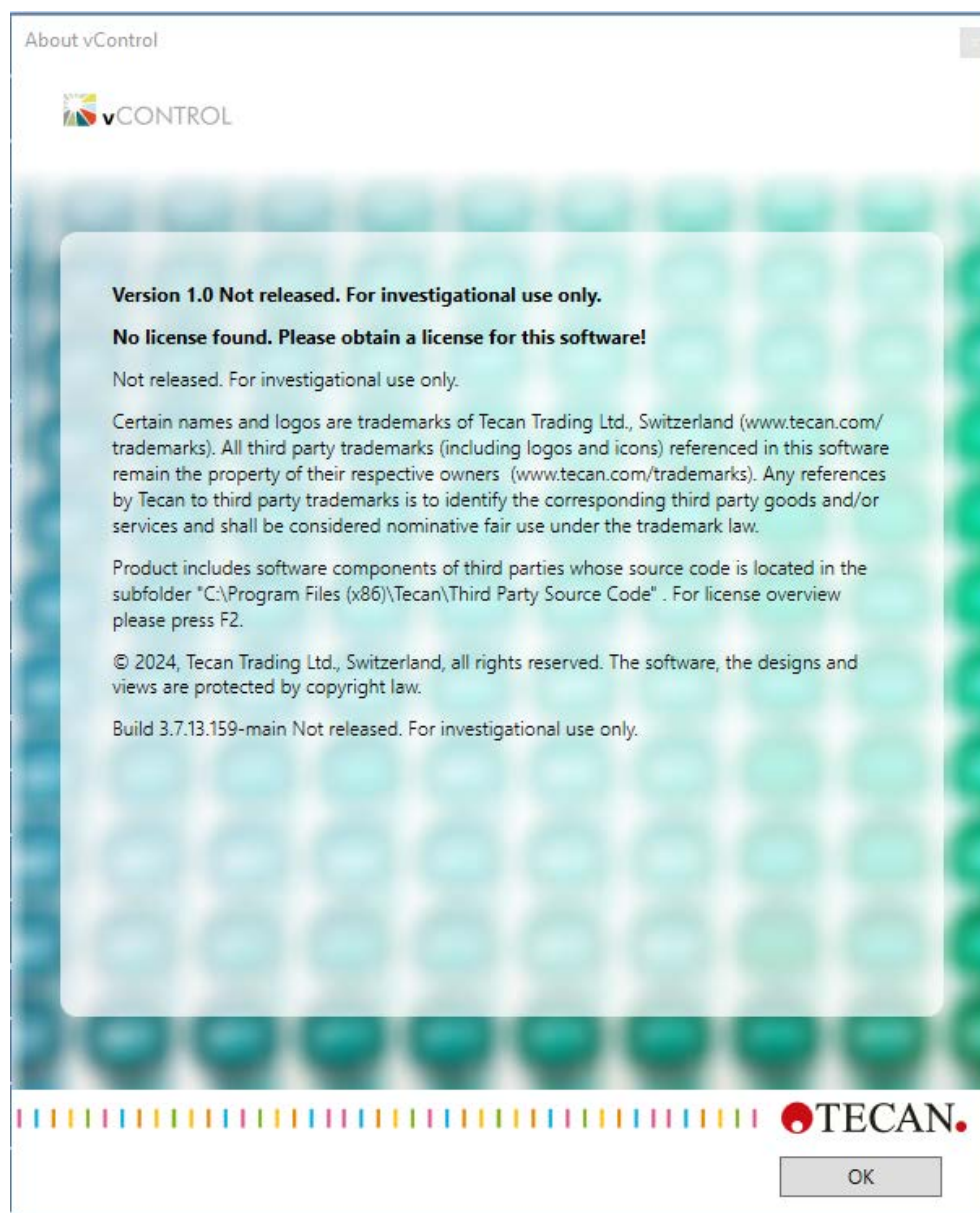


Fig. 49: About vControl

Click **About vControl** to get information about:

- Version number and Build number (in case you need to contact customer support)
- Installed Licenses
- Intended Use (The illustration above depicts an example only. The displayed intended use depends on your actual instrument and installation. For more information on the intended use of your instrument please refer to Intended Use.)
- Copyright



The product includes software components of third parties whose source code is located in the subfolder "C:\Program Files (x86)\Tecan\Third Party Source Code". For license overview please press **F2**.

7.8 Switch to Run Window Button

Switch to **Run Window** button provides a functionality to easily switch from the main editor window to TouchTools window to put TouchTools window on the front. This button automatically changes TouchTools configuration to enable "Use Full Screen" and "Use Fixed Layout".

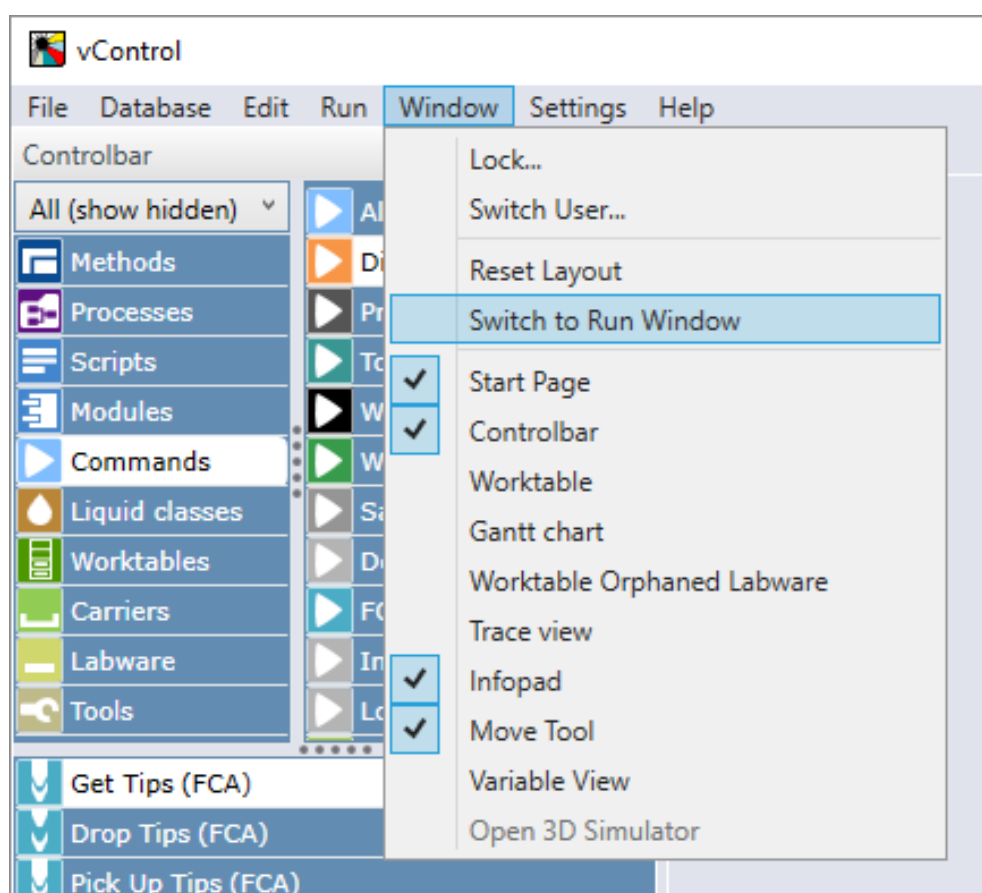


Fig. 50: Switch to Run Button

Editor Button

The **Editor Button** in Touch Tools application provides a functionality to easily switch from TouchTools to the main editor window.

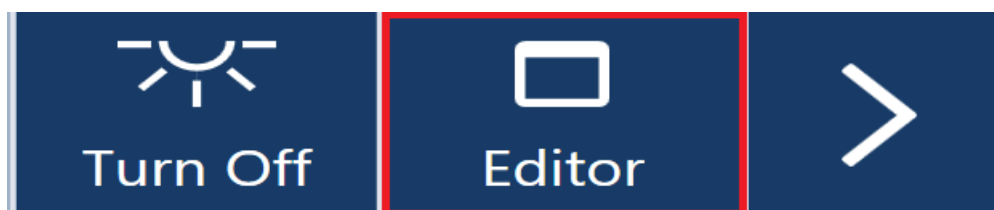


Fig. 51: Editor Button

8 Editor

The Editor allow a user to edit and adjust definitions and settings of database objects like Methods, Scripts, Processes, Worktables, Liquid Classes, Carriers or Labware.

8.1 Worktable Editor

The **Worktable** editor allows to build-up and edit a virtual worktable or parts of it. It contains of a main window with a 3D representation of the physical worktable and some navigation buttons on the right side of the window. It can be accessed via opening the worktable of interest from the Controlbar or the file menu. The worktable editor also opens automatically with the associated method, process or script when editing. All actions like placing, moving, deleting and copy-pasting carriers or labware on the worktable is done directly in the worktable editor window. Also, accessing the carrier or labware editor can be done through the worktable editor. The Infopad in the Controlbar on the left side contains additional information about a specific selected worktable object.



Fig. 52: Worktable editor

NOTICE

Equipment damage!

Incorrect segment and labware positioning on the worktable may cause arms to crash.

- Make sure the Worktable view represents all objects placed on the Veya deck.
- Only run a script if the Worktable view and the Veya deck match exactly.
 - ⇒ Only those parts of the instrument that are relevant for pipetting and carrier and labware placement are represented, but no movable parts like robot arms.

8.1.1 Worktable View

On the right side of the Worktable editor there are buttons to change the view onto the worktable and to allow navigation like rotating, zooming and pan. Two buttons allow to quickly change view perspective:

- **Default view:** Show the top view
- **Natural view:** Get the oblique projection



Fig. 53: Natural view

Move buttons allow to change worktable perspective by rotating, linear movement or zooming. Using the mouse, click the following buttons:

- **Left click:** Select object
- **Right click:** Display the shortcut menu for the active object
- **Left click & drag:** Move object
- **Right click & drag:** Rotate view
- **Left + right click & drag:** Pan view
- **Scroll wheel:** Zoom in & out

8.1.2 Worktable Overview and Labware Details

An overview of all carriers and labware on a worktable can be accessed in the Infopad under the tab "Worktable Overview". Selecting a worktable object in the list highlights the object in the window and vice-versa. This list view also allows to change the label for any worktable object, turn off visualizing the labels in the worktable window and toggling between barcodes and labware labels. The tab "Labware Details" contains further details about the selected worktable object like labware type, vendor, part number etc.

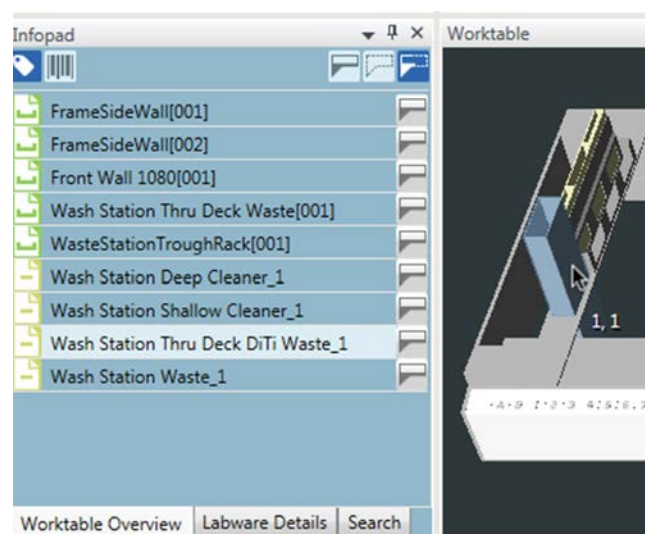


Fig. 54: Worktable Overview: Wash Station Thru Deck DiTi Waste 1

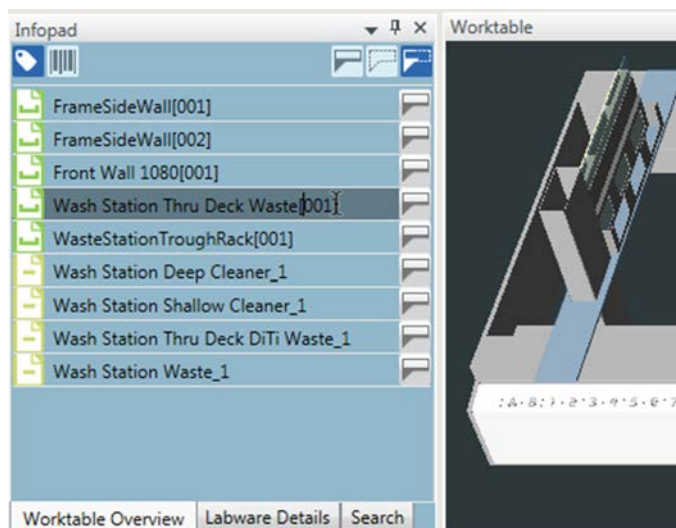


Fig. 55: Wash Station Thru Deck Waste 001

8.1.3 Base Worktable, Instrument Worktable and Script Worktable

There are three different levels of worktable layers in the software, the Base Worktable, the Instrument worktable and the Script Worktable. The Base Worktable is the one that opens when accessing the worktable editor by directly opening the worktable. It contains static objects like carriers, devices, adapters, wash stations and waste bins, all objects that will not be removed on a daily-basis from the worktable (unless for cleaning purposes). This worktable is usually used in different protocols and hence shared across different Scripts and Methods. The Script worktable in contrast is accessed when opening a script. The worktable additionally contains non-static objects like disposable tips and plates and tubes and potentially troughs. A script worktable builds on top of a base worktable and is always specific to a Script and is not shared between different protocols. The instrument worktable is explained in the next chapter.

8.1.4 Instrument Worktable

The instrument worktable is a separate worktable layer which is below the base and the script worktable. That means the base and script worktable build upon the instrument worktable. The instrument worktable itself is hidden to the user and can only be edited through the Worktable Editor while editing a worktable, script, or process. There is only one instrument worktable that gets used by all other worktables in the software. Any object placed onto this layer will appear on all other worktables in the software.

Exporting the Instrument Worktable

Whenever a worktable is added to an export, the Instrument Worktable is automatically exported together with the worktable, but not shown in the list of exported objects.

If the worktable is exported with dependencies by clicking the option **Add with Dependencies**, the objects referenced on the Instrument Worktable are explicitly added to the export window and are visible in the export. If the option **Add without dependencies** is selected, the Instrument Worktable objects are not added.

Importing the Instrument Worktable

When a worktable is imported, a checkbox **Overwrite instrument worktable** becomes visible. Enabling this checkbox will overwrite the current Instrument Worktable on the system with the one from the export file.

If the option **Add with dependencies** is selected to import the worktable, the objects referenced on the Instrument Worktable are explicitly added to the import window and are visible in the import. If the option **Add without dependencies** is selected, the referenced objects are not added.

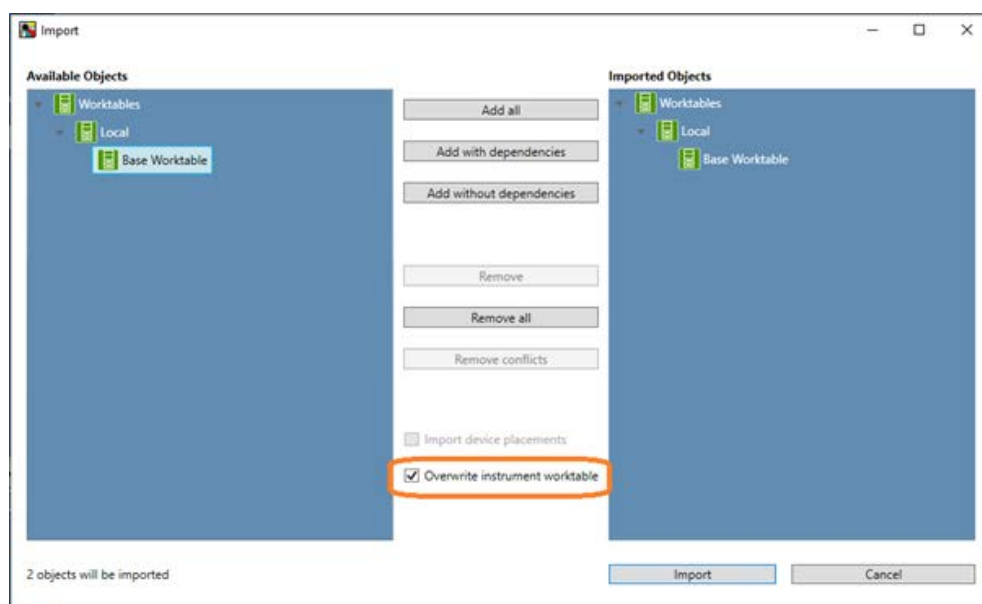


Fig. 56: Import of an Instrument Worktable



There is no necessity to make use of the Instrument worktable. If the user leaves the Instrument worktable empty and untouched, there will be no change to the previous behavior.

8.1.5 Putting Objects onto the Worktable

The Worktable can be edited by dragging and dropping objects onto it. By selecting a carrier, or a labware or tool, the positions that can be occupied by this object are highlighted in green as shown in the illustration below.

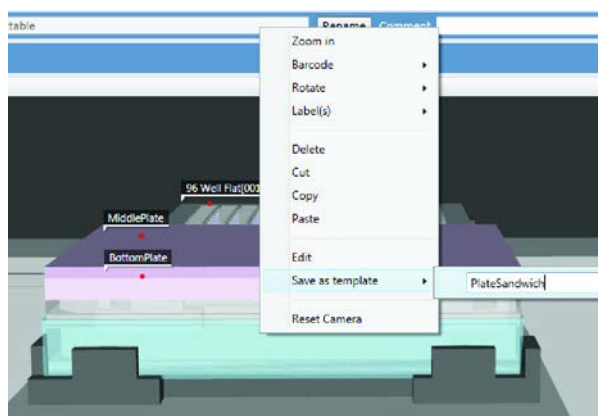


Objects can also be pre-assembled (for example carriers) and can be adapted. For example, "nests" (which are structures to hold SBS labware) can be removed from carriers. In principle, carriers can be put onto carriers and labware can be stacked. What can be put where is defined in the carrier and labware settings. For related information, refer to section Carrier Editor.

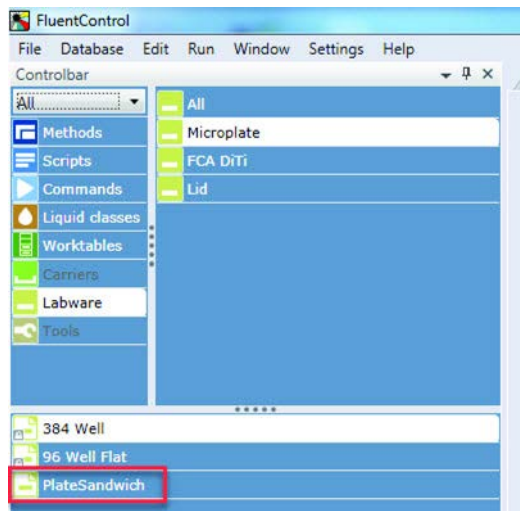
1. To create a stack of labware, place new labware on the CoverSite (see red arrow) of the labware underneath as shown below.



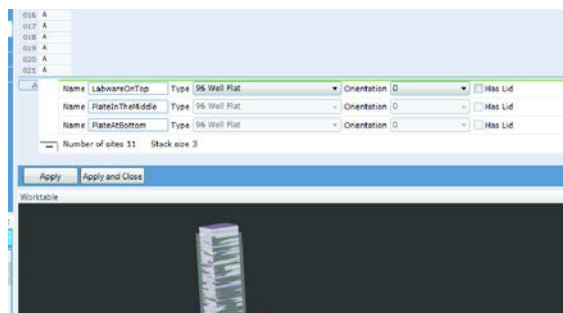
2. To save the stack of labware as a template, start by setting up the stack of plates on the worktable.
3. Right-click the labware at the bottom of the entire stack.
4. Select **Save as template** and give it a name.



5. The entire stack is now available in the Controlbar.



6. To generate plates in a storage carrier (e.g., hotel) in the Labware Group Editor, either generate the stack as a template, or create stacked plates in a storage slot by dragging and dropping plates on top of others in the same plate group.



8.1.6 Moving Objects in between Worktable Layers

Labware items can be moved in between the instrument and the base worktable when the worktable editor is open. Items can be moved between the **Script**, **Instrument Worktable**, and **Base Worktable**, if the **Script** or **Process Editor** is open. This can be done by right clicking the item which shall be moved to the respective worktable.

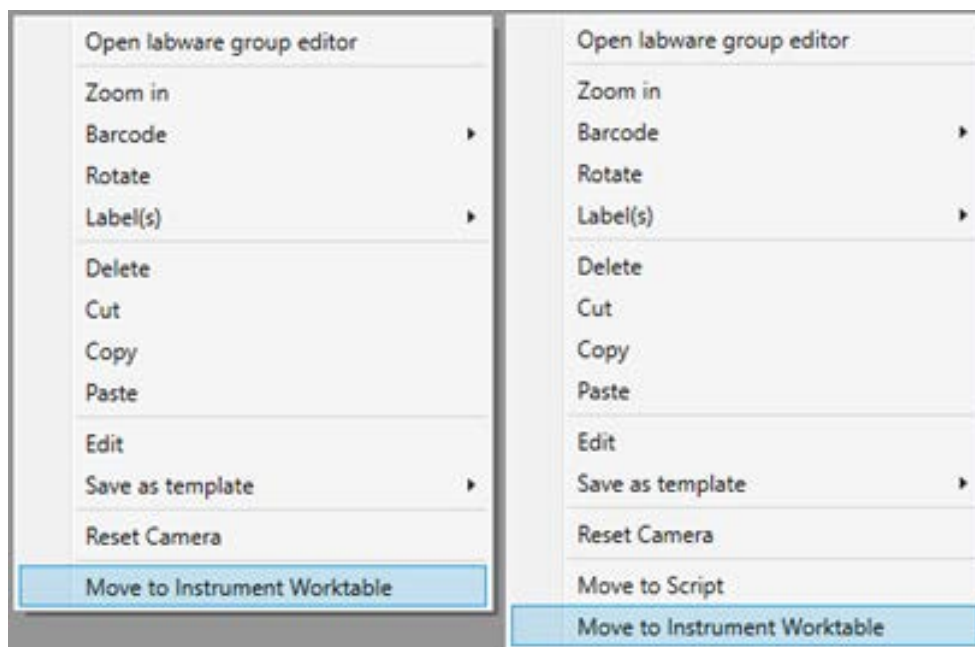


Fig. 57: After right click on in item it can be moved to the respective worktable

It is not possible to move labware to the **Instrument Worktable** when the carrier below is on the **Base** or **Script Worktable**. The option is disabled in that case. It is only possible if all underlying carriers are moved to the **Instrument Worktable** first.

8.1.7 Loading the Instrument and Base Worktable

Whenever a worktable is loaded, it will be merged with the **Instrument Worktable** present on the system. If there are no conflicts between the two worktables (e.g., two different labware items occupy the same site) then the worktable will load without issues. If conflicts do occur, the **Instrument Worktable** will be prioritized over the **Base Worktable**. The items which could not be placed onto the worktable will be shown as orphaned labware and can then be manually placed on or removed from the worktable.

8.1.8 Delta Level Setting of a Carrier or Labware Item

It is possible to define on which worktable a carrier or labware item shall be placed when it is dragged into the worktable editor. This is done by updating the custom attribute **WorkspaceDeltaLevel**

Custom Attributes		
Attribute Name	Data Type	Attribute Value
WorkspaceDeltaLevel	Integer	-2

Fig. 58: WorkspaceDeltaLevel can be edited under Custom Attributes of the respective object.

Updating the delta level to the following values will place the item onto the corresponding worktable. Items which are already present on the worktable will keep their delta level and will not be updated retrospectively.

Tab. 2: Values for WorkspaceDeltaLevel and their meaning

Delta Level Description	Delta Level Value
Script Worktable	1
Base Worktable	0
Undefined	-1 (default)
Instrument Worktable	-2

Saving a worktable when a labware is placed on a carrier that is on a higher layer is not possible and an message is displayed as shown below. The affected labware will automatically be placed on the base worktable when confirming the dialog, otherwise the user can move the affected labware by himself if the save operation is canceled.

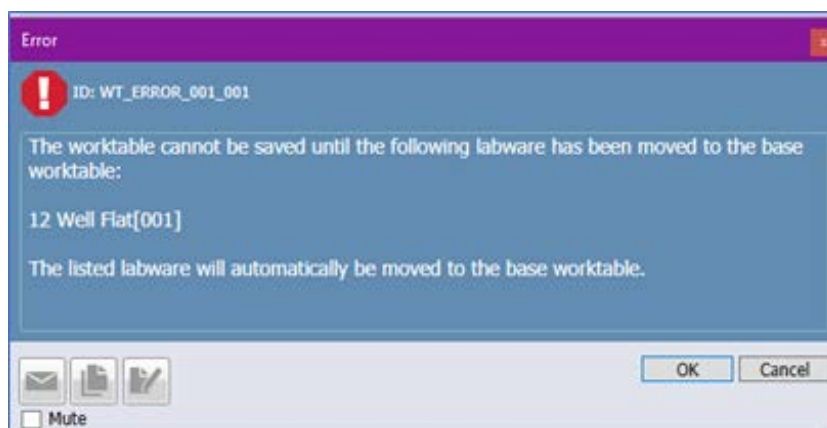


Fig. 59: Error message when trying to save a worktable with incorrect WorkspaceDeltaLevel values

8.1.9 Interconnections

The worktable editor is visible when any of the below are selected:

- **Script** editor
- **Carrier** editor
- **Labware** editor
- **Labware group** editor

8.1.10 Labware Group Editor

You can create, modify, and delete labware in the Labware Group Editor. The software provides a labware generator which will be used to specify the type and quantity of labware stacks on the worktable.

1. Drag and drop the labware on to the storage carrier to open the Labware Generator.
2. The Labware Group Editor opens.

3. Select the number of plates and the slot. White color represents occupied, blue color represents empty slots.
4. Enter a description for the assigned plate.

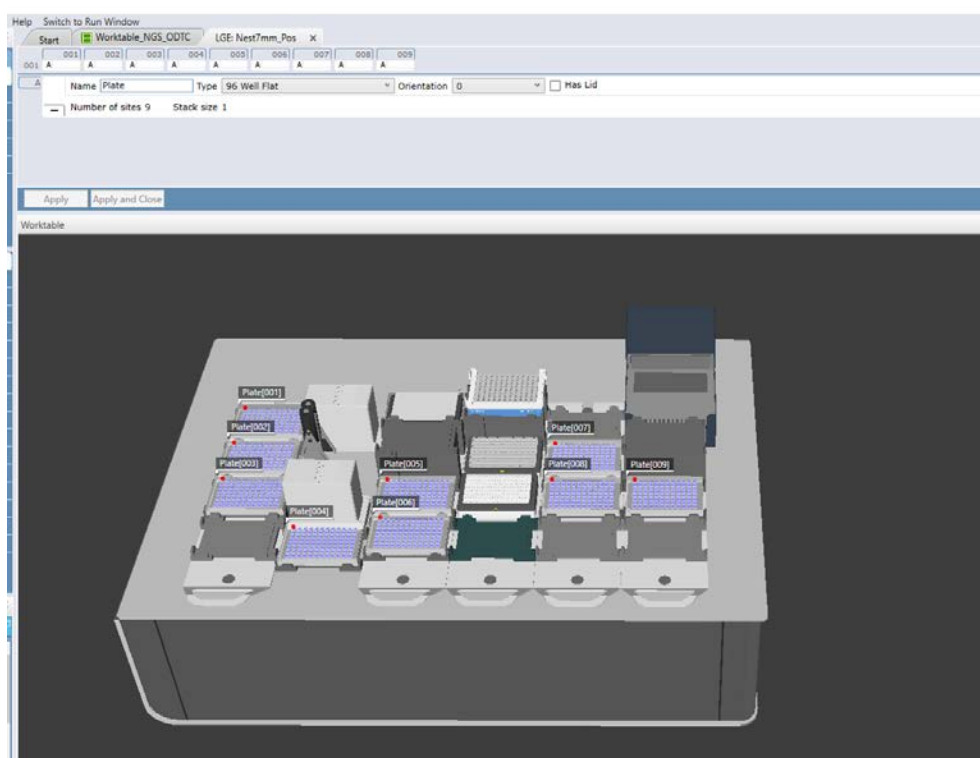


Fig. 60: Labware Group Editor

To create new labware with the Labware Group Editor, drag and drop the desired labware from the Controlbar on the left to the carrier. The Labware Group Editor automatically opens.

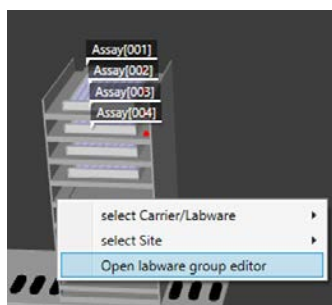


Fig. 61: Labware Group Editor / Carrier/Labware

Click the carrier sites in the graphic to create labware for the selected group. The selected labware group is highlighted in white below the list. If multiple carriers of same type exist on the worktable, the graphic displays multiple towers of carrier sites. Click to select individual carrier sites, or click and drag to select multiple carrier sites.

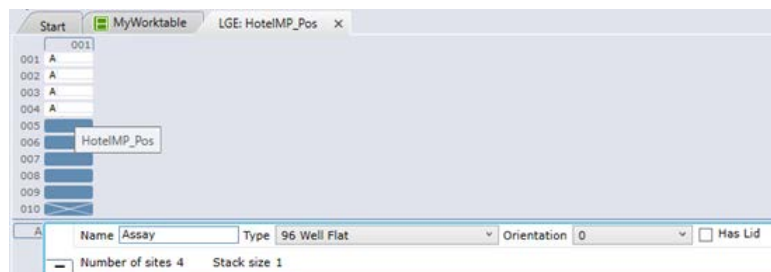
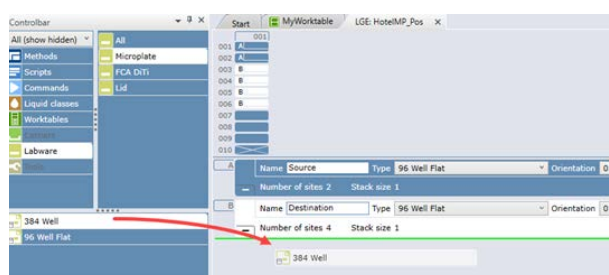
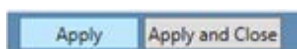


Fig. 62: Labware Hotel Slots

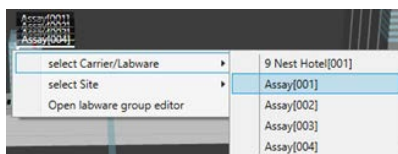
Assign a plate group name; the index numbers are generated automatically (i.e., 4 plates with the group name “Assay” results in 4 plates named Assay[001] - Assay[004] on the worktable). It is also possible to indicate the orientation of 0°/ 180° and whether or not a labware is covered with a lid. Note: For Veya stacker towers, site number 1 displayed at the top of the graphic corresponds to the plate that comes out of the stacker first. To create different labware groups, drag and drop more labware to the list at the bottom. Each labware group is flagged as A, B, C, etc., as shown in the top of the graphic.



Click “Apply” or “Apply and Close” to create the labware, which is directly visible in the Worktable window. Clicking “Apply and Close” closes the tab automatically after creating the labware.



To select labware while editing a script, click on the carrier and then select the labware from the list that appears.



8.2 Carrier Editor

View or edit carriers with this editor. It can be accessed by right-clicking on the selected carrier in the Controlbar -> Open or in the worktable editor -> Edit, via the file menu or from the Tab **Labware Details** in the Infopad -> Edit

8.2.1 General Settings

It is structured into different sections, where each section can be expanded separately. Those different sections are explained in the following sub-chapters. The General Settings sections is a collection of settings for robot-specific actions, visualization, categorization and carrier informations.

For related information, refer to section Available Drivers Enabler.

Fig. 63: General settings

Carrier name: Display name for the carrier.

Carrier is device and Device:

1. If you select the **Carrier is device** check box, you can click the **Device** arrow to see more options.
2. You can select the device type from the list.

If your device is not on the list yet, enable the device driver first.

Fig. 64: Select Carrier is device

Accessible for pipetting: The **Accessible for pipetting** check box indicates that in these positions the labware should be accessible to the FCA pipetting arm. This is primarily used for the logic of the Automatic DiTi Handling commands. Select this check box if the carrier can be accessed by a FCA pipetting arm to mount disposable tips (DiTis).



Clear the **Accessible for pipetting** check box to let the **Replace Used DiTi Racks (RGA)** command automatically replace used racks from this carrier. The **Replace Used DiTi Racks** command will only use labware placed on carriers that are not

accessible for pipetting when searching for new (full) DiTi racks. If no full racks with the specified DiTi type are found on carriers that are not accessible for pipetting, the command will only remove the old (empty) DiTi rack without replacement.

Color: If you like to use a color to make it easier to identify this carrier in the Worktable Editor, click the Color arrow to see more options. Colors only impact meshes that do not have embedded colors. They do not impact textures.

Mesh or Texture File: A mesh or texture file is used for the visualization of the carrier in the **Worktable** editor. Click Browse and select an existing mesh file or click Import to select texture file(s) to import (X3D graphics or .JPG images).

Vendor Name: You can enter the name of the carrier vendor.

Part Number: You can enter a part number for the carrier.

Link to: You can enter a link for the carrier type. The link may reference a webpage or a file and is displayed in the Infopad when selecting the carrier type. Click **Go to** to test the specified link.

Barcode: This field is currently not used by the software.

Sort in Carrier group: You can select the group in which to display the carrier in the Controlbar. Click the **Sort in Carrier group** arrow to see more options. To add a group to the list, click the ... button, then enter the new group name, and click



8.2.2 Dimensions

This section is to define the size, dimensions and bounding boxes (the no-go zones) of the carrier.

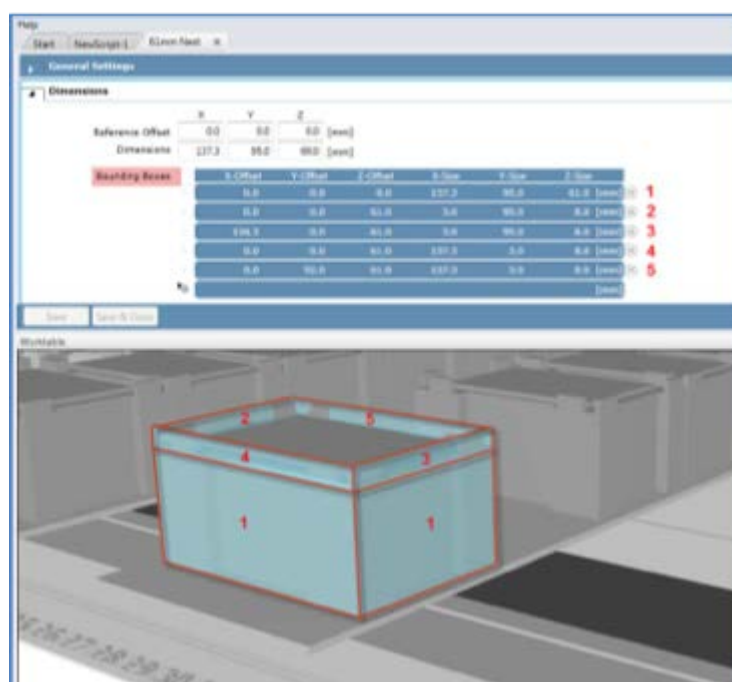


Fig. 65: Dimensions: Definition of bounding boxes that define an object.

Reference Offset: Defines the offset to the underlying carrier site and is measured in mm. The X reference offset is the distance between the left edge of the carrier and the left edge of the grid or site that the carrier is sitting on. The Y reference offset is the distance between the front edge of the carrier and the front edge of the grid or site that the carrier is sitting on. The Z reference offset is the vertical distance between the top surface of the worktable or site the carrier is sitting on and the top of the carrier.

Dimensions: Specifies the outer carrier dimensions in x, y and z measured in millimeters.

Bounding Boxes: Defines a set of bounding boxes representing the carrier outlines. Each bounding box definition consists of the dimensions and the offset from the carrier origin all measured in millimeters. Bounding Boxes are presented as a table with the following columns: **X-Offset, Y-Offset, Z-Offset, X-Size, Y-Size, Z-Size**.

8.2.3 Placement of Non-Grid-Based Devices

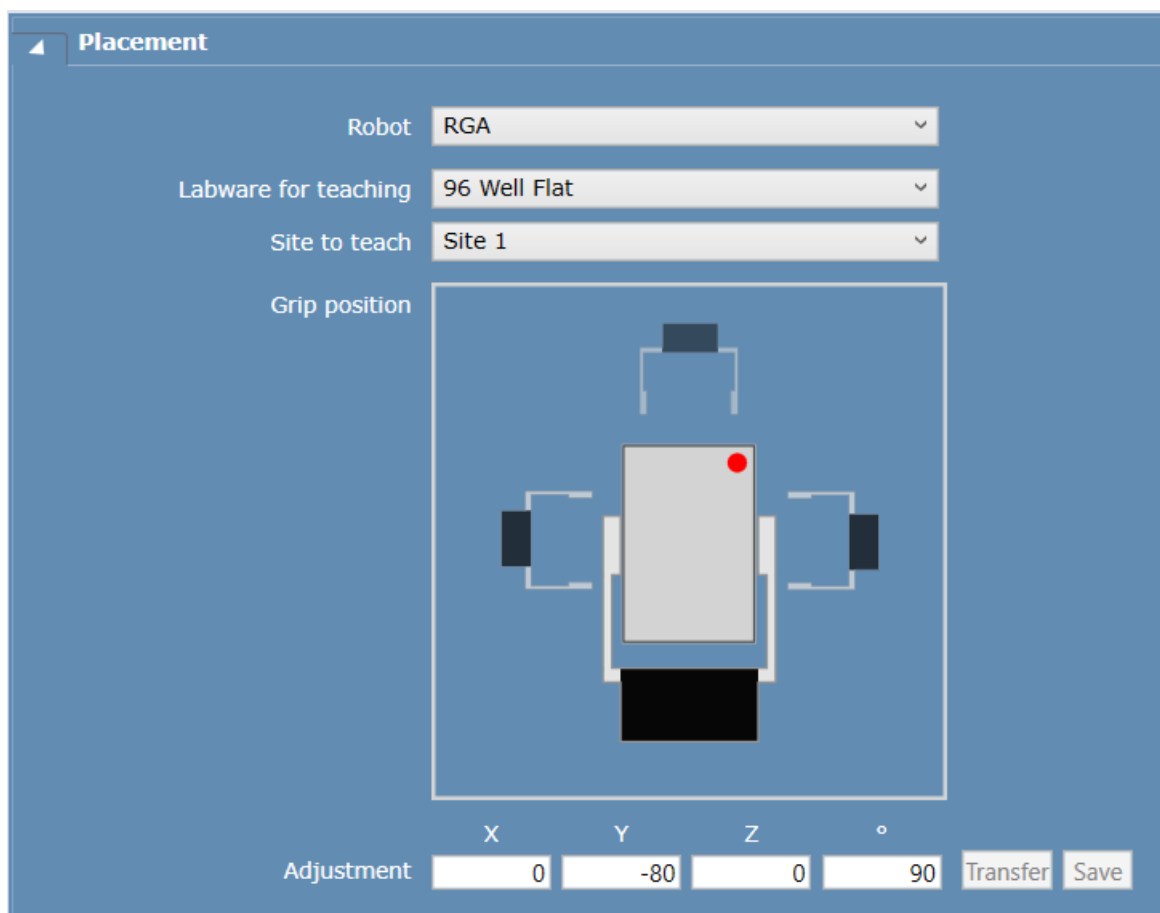
Devices such as shakers and incubators are usually integrated directly on the worktable (grid-based), while devices such as readers and washers are often integrated outside the worktable (non-grid-based), either next to or underneath the worktable. The Placement section is not present by default in the Carrier Editor, but will only appear automatically when a non-grid based device is placed on the worktable. Placement then happens via manual teaching the global coordinates of a plate sitting on its end position in the device as described in detail further below.

Robot: Defines the robot arm to use for teaching from a list of available robotic arms.

Labware for teaching: Defines the labware type to use for teaching from a list of labware types allowed on the device.

Site to teach: Defines the access site of the device, in case the device has more than one access site.

Grip position: The grip mode defines how the robot arm is approaching the labware. You can select the grip mode of the robot arm during teaching, by clicking on the appropriate robot arm picture in the Grip position figure. The figure shows the relationship of well (red dot) **A1** to the position of the fingers (grip mode).



Prerequisites for Non-Grid-Based Carrier Definitions

- Custom attribute **IsGridBased** is unchecked.
- Reference offsets are set to zero.
- “Allowed Locations the deviceSite 01-50” must be checked.

Teach Location and Orientation of the Device Carrier

The Placement section opens up when you integrate a non-grid based device. Refer to section [“Fetch Labware for Teaching \(RGA\)” \[175\]](#).

1. Start the vControl application.
2. Fetch labware for teaching via a direct command.
3. Drag and drop a non-grid-based carrier, such as a reader, on to the worktable. The **Placement** section in the device editor opens.
4. Activate **Zero-G** for the RGA and move it to the end position on the plate holder of the device.
5. Click **Transfer** under **Adjustment**.
6. Click the Save button next to the Transfer button. As a result, the positioning of the device on the worktable updates and the device placement is stored on the connected instrument.

7. Go to the **Robot Vector** section and transfer the current position also as the **End Position** of the vector.
8. Teach safe and intermediate waypoints for RGA vector.
9. Return the labware for teaching via a direct command.

Placement only teaches and defines the location of the device in the workspace. A device-specific vector must still be manually defined in the **Robot Vectors** section. It is important that the **Adjustment** in the **Placement** section is the same location as the **End Position** of the vector in order to prevent Pathfinder issues.

The Pathfinder does not work underneath the deck. Therefore, the first waypoint of the RGA vector (the safe position) must be defined above deck level.

8.2.4 Site Definitions

A site is a defined location, where another carrier or labware can be placed onto. Properties of the sites are defined in the site definition window, including number of sites per carrier and their sizes as well as which kind of vectors are allowed as well as which grip modes are allowed to move the objects on the sites.

By default, the **Transfer Labware** command supports all grip modes. The priority however is given to the narrow vector. If you want to change this setting, deselect any grip modes that you do not want to be used for that particular site. The site coordinates are relative to the carrier's 0 point defined by its reference offset.

All dimensions in the table are specified in [mm]. The rotation is specified in [°].

Select robot: Click the **Select robot** arrow to see more options. The instrument must be equipped with at least one RGA otherwise the list is empty.

N, NR, W, WR: **N**, **NR**, **W**, **WR** are called the Grip Modes: **N** = Narrow, **NR** = Narrow Reverse, **W** = Wide, **WR** = Wide Reverse. Check marks in a Grip Mode indicate it is allowed at that site.

Site Definitions

Positions: 7

All dimensions in the table below are specified in [mm].
The rotation is specified in [°].

Grip Modes: N = Narrow, NR = Narrow Reverse, W = Wide, WR = Wide Reverse
Select robot: [dropdown]

Site	X-Offset	Y-Offset	Z-Offset	X-Size	Y-Size	Z-Size	Rotation	Location	N	NR	W	WR
1	6.35	517.50	4.00	137.30	95.00	0.50	0.0	FCA_DITWest_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	6.35	421.50	4.00	137.30	95.00	0.50	0.0	FCA_DITWest_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	6.35	325.50	4.00	137.30	95.00	0.50	0.0	FCA_DITWest_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	6.35	229.50	4.00	137.30	95.00	0.50	0.0	FCA_DITWest_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	6.35	133.50	4.00	137.30	95.00	0.50	0.0	FCA_DITWest_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	6.35	37.50	4.00	137.30	95.00	0.50	0.0	FCA_DITWest_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	-17.50	-136.90	-602.70	185.00	110.00	606.70	0.0	WasteSide_Site_1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fig. 66: Site definitions

Site Definitions are presented as a table with the following columns:

Position Layout: Defines how the layout of the different sites are. For a tube runner with 24 tube positions the layout would be XS: 1, YS: 24, ZS: 1. For a SLAS-format tube carrier with 6x4 tubes the layout would be more like: XS: 6, YS: 4, ZS: 1

- **Site:** Full integer number. There cannot be two sites with the same number.
- **X-Offset:** Offset in X from the carriers 0-point.
- **Y-Offset:** Offset in Y from the carriers 0-point.
- **Z-Offset:** Offset in Z from the carriers 0-point

- **X-Size:** Usually as big as the footprint of the labware/carrier that will be placed onto this site.
- **Y-Size:** Usually as big as the footprint of the labware/carrier that will be placed onto this site.
- **Z-Size:** Usually small. Needs to be large enough such that the site can be seen in the worktable window and selected.
- **Rotation:** or landscape format this is 0.
- **Location:** The name of the site itself. If multiple sites share the same location group name then.
- **N**
- **NR**
- **W**
- **WR**

Positions: Specify how many sites the carrier has.



If multiple carriers are stacked (e.g. Nests on Base Plates), the Grip Mode used will be the Grip Mode that is allowed by all carriers in the stack. For example, site 1 of the base plate may only have Wide selected. The Nest may have all 4 selected. Transfer Labware will only use the Wide vector at this site as it is the only one that is enabled for the base plate and nest.

The selection of the vector occurs before the Path Finder is asked to move an arm. Potential collisions or impossible moves are not considered during this determination.

Any given layer that has no selected Grip Modes will not impact the determination of the allowed Grip Modes.

If a source position and a target position of a Transfer Labware command do not share allowed Grip Modes, a Regrip Nest which allows both Grip Modes must be present somewhere on the worktable.

You can specify the size and positions of the sites in x, y and z. It is also possible to group the sites by editing the Location name. You can specify the dimensions and offsets for each of the sites of the carrier.

8.2.5 Robot Vectors



*Robot vectors are **NOT** checked for collisions. Only if the transfer from or to the safe position takes place via the pathfinder is this part checked for collisions.*

This is the vector teaching section of the **Carrier** editor.



Robot Vectors

Carrier currently selected for teaching: Infinite F50[001]
RGA is currently selected in the Move Tool.

Location: InfiniteF50_Pos Position: Site 1
Robot: RGA Vector: Narrow

Adjustment: X: 0 Y: 0 Z: 0 [mm] Show delta Save
☐ Site definition in carrier template
☐ Individual position of location in this workspace

The adjustment only affects the site currently selected for teaching.
Site adjustments in the carrier template will only be effective in this workspace after saving.

Define Coordinates							
X [mm]	Abs	Y [mm]	Abs	Z [mm]	Abs	Rotation Z [°]	Abs
82.2	<input checked="" type="checkbox"/>	-27.6	<input checked="" type="checkbox"/>	98.0	<input checked="" type="checkbox"/>	90.0	<input checked="" type="checkbox"/>
82.2	<input checked="" type="checkbox"/>	142.4	<input checked="" type="checkbox"/>	98.0	<input checked="" type="checkbox"/>	90.0	<input checked="" type="checkbox"/>
82.2	<input checked="" type="checkbox"/>	142.4	<input checked="" type="checkbox"/>	86.0	<input checked="" type="checkbox"/>	90.0	<input checked="" type="checkbox"/>
	<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>		<input type="checkbox"/>

Transfer Move to selected Move to first Move to last

Grip Parameters (for teaching purposes only)

Grip Distance: 75 [mm] Grip
Release Distance: 88 [mm] Release
Force: 30 [%]

Fig. 67: Robot Vectors of the 7 mm Regrip Nest


Location: A location is a logical group of sites that can split a carrier or combine sites. Click the **Location** arrow to see what locations are available for teaching robot vectors. Select the entry from the list that you want to use.

Position: A position is a 3-dimensional numbering of positions within a location. Click the **Position** arrow to see what positions on the carrier are available for teaching robot vectors.

Robot: Click the **Robot** arrow to see more options. The instrument must be equipped with at least one Robotic Gripper Arm (RGA), otherwise the list is empty.

Vector: Select which Grip Mode vector shall be taught for the selected robotic arm. Click the **Vector** arrow to see more options.

The following default vectors are available for teaching:

- **Narrow:** Labware gripped in portrait orientation.
- **NarrowReverse:** Same as Narrow but Reverse = gripper rotated by 180°.
- **Wide:** Labware gripped in landscape orientation.
- **WideReverse:** Same as Wide but Reverse = gripper rotated by 180°.
- **Custom vectors:** You can teach custom vectors. Click  to add a new vector name.

Adjustment: It is possible to adjust the vectors for each site of a carrier, for example if one of the sites deviates in shape from all others. Two ways of executing the adjustment are possible:

- To change the values for the selected site in all carriers of this type, click the Site definition in carrier template radio button.



Note that with this option, this adjustment is stored only in the local database, not on the connected instrument.

- To change the values for the selected site only in the current worktable, click the Individual position of location in this workspace radio button. With this option, the adjustment is lost when the instance of this carrier is removed from the worktable. With this option, the site adjustment will be stored on the connected instrument (as described in chapter 9.1 - Teaching) and will be applied to all worktables that have the same type of carrier at the same location.



Adjustments applies only to the site which is selected for teaching.

1. To update the values, click **Show delta**.
The adjustment is displayed in the X, Y, and Z boxes in [mm].
2. To apply the new values, click **Save**.

Define Coordinates: The robot vector is displayed as a list of coordinates in the **Define Coordinates** table. The table that represents coordinates has the following columns: **X** [mm] / Abs, **Y** [mm] / Abs, **Z** [mm] / Abs, Rotation **Z** [°] / Abs.



Vector positions are displayed relative to the site and will not match the Move Tool coordinates, which are absolute. Vector positions are plate based, not tool or arm based.



It is important that the last coordinate in the list of the default vectors has a relative value of almost 0 for X,Y and Z to avoid pathfinder issues.


Possible reasons why the value differs from 0 after teaching: - Wrong site definitions of the current carrier or of the deck/grid segment the carrier is placed onto. - Non-grid-based device was not taught properly or needs to be re-taught because it moved slightly because of vibrations.

Now, select a row, you want to edit.

Transfer: To include the current coordinates of the robotic arm in the **Define Coordinates** list.

1. Click **Transfer** and let vControl do the math for you (convert the actual position to a relative value to be stored).

You can update coordinates of the coordinates list, or you can add new coordinates at the end of the list.

2. To add new coordinates, click , select the blank row at the end of **Define Coordinates**.

3. Click **Transfer**.

For convenient teaching, use the robotic arm in Zero-G mode and move it by hand into the desired position.

Alternatively, you can move the arm with the help of the buttons provided by the Move Tool or the numeric keypad.

Move to selected: To move the robotic arm directly to the selected coordinates (an entry in the Define Coordinates list above), click **Move to selected**:

- No collision checking is done!
- Use **Move to selected** only if the robotic arm is near the selected coordinate.
- Make sure that there are no obstacles in the way!

Move to first; Move to last; Move to selected: To move the robotic arm to the topmost entry on the **Define Coordinates** list, click **Move to first**. Path Finder will be used, if the robotic arm was not moved to any of the other coordinates, thus avoiding collisions with worktable objects. If the robotic arm was moved to any entry on the list before, the arm will move directly, without Path Finder support. It will move along each entry on the list and finally to the coordinates of the topmost entry. Use **Move to first** in combination with **Move to last** to test the whole robot vector.

To move the robotic arm to the coordinates at the end of the **Define Coordinates** list, click **Move to last**. If the robotic arm was not moved to any other entry on the list, the arm will first move to the coordinates of the topmost entry using Path Finder thus. After that, it will move to the coordinates of the next entry without Path Finder support until the coordinates of the last entry are reached. If the arm was moved to another entry on the list beforehand, it will move directly, without Path Finder support. The arm will move along each entry on the list until the coordinates of the last entry are reached.

If you use **Move to selected**, **Move to first**, or **Move to last** with the door open, the arms will move slowly. The arms will move at full speed if the safety shield is closed.

Grip Parameters (for teaching purposes only)

Use the Grip Parameters (described below) for gripping and releasing labware while teaching. The gripper width is not stored as part of a robot vector. The commands open the grippers automatically to the necessary width. But while teaching, the user is responsible for opening the grippers to an appropriate width. Otherwise, the grippers might collide with the target carrier or labware placed there.

These settings are for teaching only and will not be used in the script commands. The script commands will use the values stored in the labware that is moved.

Grip Distance: To move the fingers from the current position to this position unless the grip force is reached in between, click **Grip**.

Release Distance: To move the fingers from the current position to this position, click **Release**.

Force: Specify the grip force relative to the grippers capabilities that will be reached when gripping.

⚠ CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

- ✓ You can store several robot vectors per carrier.
- ✓ Open the gripper to an appropriate width before moving.

1. Select a carrier on the worktable to be used for teaching.

The text at the beginning of the **Robot Vectors** section tells you the carrier currently selected for teaching, otherwise the following phrase is displayed: Please select a carrier on the worktable to be used for teaching

2. Select a robotic arm in the Move Tool.

The second sentence tells you the robotic arm currently selected in the Move Tool.

8.2.6 Automatic Site Adjustment

⚠ CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

☐ **Automatic Site Adjustment**

Carrier currently selected for teaching: 7mm Nest[001]
Please select the robot for teaching below.

Robot FCA

Adjustment X Y Z 0 0 0 [mm] Teach Apply

The adjustment only affects the site currently selected for teaching.

Fig. 68: Automatic Site Adjustment

To measure the deviation between the physical and the nominal position of a nest automatically, a Teaching Plate has to be placed to it. The related labware definition can be found in the Labware -> Miscellaneous group in the Controlbar.

Automatic Site Adjustment can be performed with a Reference Tip, Fixed Tip or a Disposable Tip whereby the accuracy decreases with the order of that listing.

To mount and unmount a Reference Tip, execute the commands as described in [“Manually Mount Tool \(FCA\)” \[▶ 175\]](#) and [“Manually Unmount Tool \(FCA\)” \[▶ 175\]](#). To pickup and drop a Disposable Tip, execute the commands as described in chapters Get Tips (FCA) and Drop Tips (FCA).

Open the Carrier Editor of the nest you want to teach. Note: Automatic Site Adjustment allows teaching also on Carrier level. But it's strongly recommended to teach every nest individually as the measurement of a single point is not accurate enough to cover the deviation of a whole carrier.

The nest selected for teaching is displayed at the top of the section. Robot: The selected Robotic Arm that should perform the measurement.

Once the desired Teaching Tool is mounted and the Teaching Plate is placed on the nest, the parent site of the nest can be taught by clicking on the Teach button. The arm will approach the first accessible teaching triangle pair as defined in the Custom Attributes of the Teaching Plate and executes a capacitive scanning in X, Y and Z direction. The result will be shown in the boxes next to it.

The measured deviations can be stored by clicking on the Apply button.

8.2.7 Allowed Labware Types

This section contains information which labware is allowed on the carrier. You can select the labware types which may be placed on the currently edited carrier. This is what determines the highlighted areas in the worktable editor while placing labware.

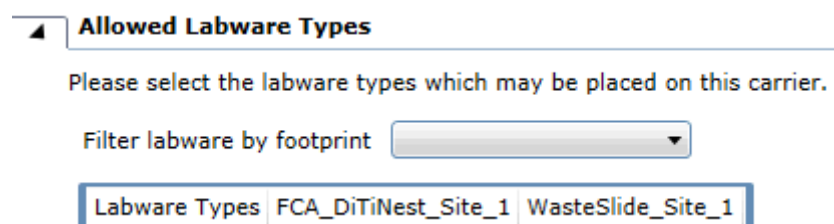


Fig. 69: Filter labware by footprint

Filter labware by footprint: You can specify the footprint of the labware. Click the **Filter labware by footprint** arrow to see more options. The list contains the different footprints of all labware types present on the system.

8.2.8 Allowed Locations

This section contains information on which location the carrier is allowed, including worktable grids or other carriers. You can select the locations on which the currently edited carrier type may be placed.

Fig. 70: Filter locations

Filter locations: Click the Filter locations arrow to see more options. The selection made in the **Filter locations** list (e.g., <all selected>) has an influence on the **Allowed locations** list below.


Allowed locations: The **Allowed locations** list shows on which location the carrier or labware is allowed. Each entry comes with a check box, which you can clear or select. To narrow down the list, use **Filter locations**.

8.2.9 Custom Attributes

You can define custom attributes as parameter on carrier or labware to influence specific system (also robot) actions.

Attribute Name	Data Type	Attribute Value
WorkspaceDeltaLevel	Integer	-1

Fig. 71: Custom attributes

- To delete a custom attribute, click .
- Click on row to the custom attributes table.
- Click the **Attribute Name** arrow to see all attributes:
 - AccessLocation:** Location to access labware for transfer from storage device.
 - BoundingBoxVisible:** Shows the BoundingBox of a carrier.
 - Grasp_location_max:** Maximum value for allowed grasp.
 - Grasp_location_min:** Minimum value for allowed grasp position range.
 - Grasp_location_N_max:** Maximal plate grip offset narrow.
 - Grasp_location_N_min:** Minimal plate grip offset narrow.
 - Grasp_location_W_max:** Maximal plate grip offset wide.
 - Grasp_location_W_min:** Minimal plate grip offset wide.
 - Grip_Max:** Maximal gripper distance for labware on this carrier. This applies for the RGA and for the FCA
 - Grip_Min:** Minimal gripper distance for labware on this carrier. This applies for the RGA and for the FCA
 - IOState:** I/O state of associate tool driver (tool location). If IOState is disabled, then the system does not communicate with the tool location driver.
 - IsFrameElement:** Set object as part of the worktable frame.

- **IsGridBased:** Define if a worktable is grid based or not.
- **Lid:** Whether a lid is allowed on the labware place on this carrier.
- **Lid_location:** Location to store the lid.
- **Lid_position:** Position of lid within lid location.
- **LiddingStation:** Carrier that is allowed to cover or uncover a lid from a labware.
- **OpenCarrierAndSiteList:** Open labware and site list.
- **OpenLabwareGenerator:** Open labware generator.
- **PassiveStack:** Carrier is treated as a passive stack.
- **RegripStation:** Location to use for re-gripping.
- **RenderVisualChildren:** Define if children objects of a worktable object shall be rendered.
- **RobotMove:** On-The-Fly Tools: To enable/disable robot move.
- **Server:** Call name of the device (internal name of driver).
- **Speed:** Maximum speed of robotic arm in percentage of full speed.
- **StartRack:** Defines first position of a group of stacks for the carousel.
- **Storage:** Defines if the lid shall remain at the location.
- **WorkspaceDeltaLevel:** Preferred workspace delta level for this labware.
- **WasteLocation:** Location is waste location.
- **MapTemplateName:** The template name in MAP
Loading ID 5-grid_TubeRunnerID: The ID of the Tube Runner

8.3 Labware Editor



Tecan endeavors to ensure the correctness of the default labware definitions delivered with vControl. Many labware definitions are based on a specific manufacturer's part that is listed in the software. However, for non-Tecan labelled labware, these dimensions can change, so you are advised to check that the labware you are using is in agreement with the labware definition and validate your process using the labware you plan to use.

If you use different labware, you must carefully check that the labware definition in the software agrees with the labware you are using. Incorrect compartment definitions can lead to air aspiration or crashes.

View or edit Labware with this editor.



Fig. 72: Labware Editor

8.3.1 General Settings

General Settings

Labware name: 96 Well Skirted PCR_1

Type: Regular

Labware footprint: Microplate

Tips per well: 1

Sort in Labware group: Microplate

Fig. 73: General settings

Labware name: Defines the display name of the labware.

Type: Click the **Type** arrow to see more options.

Labware footprint:

1. Select the item to which the currently edited labware should be mapped in **Allowed Labware Types**.
2. Click the **Labware footprint** arrow to see more options.
3. If you need another form factor, click ... to add a new item.

Tips per well: This parameter is available only in a **Labware** editor. Click the **Tips per well** arrow to see more options.

Sort in Labware group: Defines where the labware should appear on the **Controlbar**.

Tool ID

This parameter is available only in a **Tools** editor.

8.3.2 Positions / Teaching

CAUTION

Moving parts!

Moving parts can cause crushing or cutting injury.

- Always keep hands and body away from moving parts while teaching with open safety panels.
- Take unintended motion of robot arms into consideration while teaching with open safety panels.
- Do not attempt to circumvent the door sensors.

You can teach the positions of a labware on the worktable.

Positions / Teaching

Dimensions X: 127.8 Y: 85.5 Z: 14.4 [mm]

Bounding Boxes

	X-Offset	Y-Offset	Z-Offset	X-Size	Y-Size	Z-Size
	0.0	0.0	0.0	127.8	85.5	14.4 [mm]
						[mm]

Number of wells X: 12 Y: 8

Well alignment ☒ regular ☐ interleaved

Compartments per well: 1

Labware currently selected for teaching: 96 Well Flat[001]
Arm currently selected for teaching: RGA

Current compartment: 1

Pipetting positions All values are specified in [mm].

	X	Y	
(position #1)	14.38	74.24	<< Transfer Move to
(position #96)	113.38	11.24	<< Transfer Move to

(adjustment) X: 0.0 Y: 0.0 Z: 0.0

Show delta Save

The adjustment only affects the site hosting the currently selected labware.

Z-Coordinates

Z-Travel	22.40	<< Transfer Move to
Z-Start	14.30	<< Transfer Move to
Z-Dispense	12.40	<< Transfer Move to
Z-Max	4.00	<< Transfer Move to
Z-Bottom	3.50	<< Transfer Move to

Y-distance of wells

Fig. 74: Positioning / Teaching

Dimensions: Defines the outer labware dimensions in **X**, **Y** and **Z** measured in millimeters.



Bounding Boxes: A bounding box is a virtual area the pathfinder will not go into. The bounding box should reflect the physical dimensions and there can be several bounding boxes with different offsets interfering with each other in order to reflect all parts. New bounding boxes can be added pressing the + button. To remove press the x button. .

Number of wells: These two fields specify the well geometry:

(X): Specify the total number of wells in the X-direction (max. 128).

(Y): Specify the total number of wells in the Y-direction (max. 128).

Well alignment: Select the style of well alignment you want to use. Click the regular or the interleaved option button.

Type	Name	Comment
	Rectangular labware (regular)	This is labware where the wells are arranged in regular rows and columns (e.g., a standard microplate). The first well is rear left; the last well is right front.
	Rectangular labware with offset (interleaved)	This labware is similar to the rectangular type, but every second column is displaced in the +Y-direction. The spacing of the wells in the Y-direction is the same for all columns. In addition to specifying the X- and Y-positions of the first and the last well, you must also specify the Y-position of the first well in the second row (position #2).

Compartments per well: Some special labware types (e.g. for protein crystallography) have wells with several compartments (cavities). In this way it's possible to pipette different liquids into each of the compartments. Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Specify the number of compartments per well in this field. For normal labware, select 1.

Please select a labware on the worktable to be used for teaching ...All subsequent options are disabled (greyed-out) if no labware is selected in order to teach it's positions in x, y and z. To allow this, open the required labware for teaching directly from the worktable editor.

Current compartment: For labware with more than 1 compartment per well, select the Current compartment to be used for teaching.

Pipetting positions: All values are specified in [mm].



The adjustment only affects the site hosting the currently selected labware.

1. Click the Move To button to bring the channels to the current reference position in x and y.
2. Use the robotic arm in Zero-G mode and move it by hand into the desired position.

Alternatively, you can move the arm with the help of the buttons provided by the Move Tool or the numeric keypad.



First teach the X and Y adjustment. Then for Z teaching, move to Z-max first and then move down to Z-bottom with the Move Tool to avoid crashes.

3. Click <<Transfer if the labware sites shall be adjusted globally, i.e. for all sites the labware sits on.
4. If the positions shall be only taught for that specific site the labware is currently located the (adjustment) option should be used instead of the global corrections done in point 3. Press Show Delta to see the delta of the desired end position to the reference position. If this is ok, press **Save** to adjust the positioning of only this one site.
5. **Z-Coordinates:** This part defines the z-coordinates of different height within the labware. Z-Bottom refers to the inner bottom of the well relative to its reference position, which is its site (i.e. in z-coordinates it's the outer bottom of the well). Z-Max defines the lowest point the software will move to and pipette from. This includes some safety margin to the bottom and also means some dead volume. It is recommended to always have at least 0.5mm safety between z-Bottom and z-Max. Z-Dispense defines the free dispense position, i.e. a position within the well where the tip will not touch the liquid. Z-Start defines the position where the liquid level detection starts, this is usually also within the well but can also be outside of the well. Z-Travel is outside of the well and a safe position where the pathfinder can move to. This also means the value must be higher than the z-size of the bounding box.

Use << **Transfer** and **Move to** to teach the Z-coordinates corresponding to the description above.

Y-distance of wells (Visualization only):

If the pipetting operation specifies the use of more than one tip, vControl adjusts the spacing between the tips in the Y-direction to allow pipetting to take place on several wells simultaneously if possible. The minimum spacing between adjacent tips is 9 mm and the maximum spacing is 38 mm, as shown by the light green area in the middle of the indicator. vControl calculates the Y-distance (the spacing between adjacent wells) from the labware coordinates and the number of wells specified. This is shown by a vertical line which is colored either dark green or dark red. To optimize pipetting speed, the vertical line should lie within the green area. If the vertical line lies within the red area, you may have specified the coordinates incorrectly or inaccurately. Alternatively, you are using labware with a non-standard design. The yellow line shows half of the minimum Y-distance (i.e., 4.5 mm). If you are using microplates with 384 wells and choose this Y-distance, pipetting can take place simultaneously to every other well.

Labware Coordinates: You can define the following different heights (Z coordinates) for the wells.

The coordination system is defined in vControl as follows:

- Front, Left, Worktable level of the instrument is the 0,0,0 point.
- Positive X is to the right.
- Positive Y is to the back and Positive Z is Up.
- Negative values are the opposite directions.

See also The pipetting instrument's coordinate system. Labware coordinates are always relative to the carrier. Therefore, coordinates can only be edited if labware is placed on carrier. Values must not be entered manually but added by the << **Transfer** button.

The following table describes the X, Y and Z coordinates in the positions/teaching section of the **Labware** editor:

X, Y and Z coordinates for labware:

Coordinate	Position / Function
X,Y	At this position, the tip is situated above the well/tube of the first/last position on the labware.
Z-Travel	Tip height at which the tips move from one X/Y position to another. Choose Z-Travel to be higher than all obstacles on the carrier. Movements that pass over different pieces of labware use a tip height of Z-Travel (global) instead.
Z-Start	Tip height at which the liquid level detection (ILID) is switched on during pipetting. For tubes and containers (troughs), choose Z-Start to be at least 1 mm below the rim but above the liquid surface. For microplates, choose Z-Start to be at least 1 mm above the edge of the well.
Z-Dispense	Tip height at which liquid is dispensed. It should be low enough to ensure that no liquid drops can get into adjacent wells or tubes.
Z-Max	Tip height which is very slightly above the lowest point of the well, tube or container. It is the lowest position which the tip is allowed to reach without touching the bottom. When searching for the liquid surface, the pipetting instrument will search from Z-Start down to Z-Max . If the tip reaches Z-max without finding liquid, the pipetting instrument reacts according to the configured liquid level detection error mode. NOTICE! If "Aspirate Anyway" is selected as an error handling option, the tip will not track lower than Z-Max during aspiration.
Z-Bottom	It defines physical bottom used for volume calculation.

8.3.3 Compartment Definitions

The compartment definition reflects the inner geometry of one well or cavity. It is used to calculate liquid volume from a detected liquid level. Therefore in order to have an accurate liquid volume calculation it is recommended to have accurate and precise compartment definitions.

Compartment Definition: Example with 3 shapes (Cylinder, (Truncated) Cone and Partial Sphere).

Compartment Definition

Select shape

All dimensions in the table below are specified in [mm].
Square measures are defined in [mm²].

Shape	Height	Diameter	Buttons	
Cylinder	4.00	5.50	⬆ ⬇ ⬇ ⬆	
Shape	Height	Bottom diameter	Top diameter	Buttons
(Truncated) Cone	9.00	3.00	5.50	⬆ ⬇ ⬇ ⬆
Shape	Height	Diameter	Buttons	
Partial Sphere	1.50	3.00	⬆ ⬇ ⬇ ⬆	

XZ-View YZ-View

0 5.5 0 5.5 14.5

Fig. 75: Compartment Definitions

Select shape:

There are several predefined shapes provided. Depending on the selected shape different parameters are needed. Alternatively a user defined option is available too. This option allows to define a matrix of known height-volume combinations (input either directly or via a file). A cavity can be defined using more than one shape as in the example here. Use the Add button to add more shapes, the x button at the end of each table entry to remove the shape or alternatively the up and down buttons to move the entries.

Compartment Definition

Select shape

- Partial Sphere
- Cylinder
- (Truncated) Cone
- Cuboid
- (Truncated) Pyramid
- Trapezoid
- User Defined

Fig. 76: Select shape arrow

8.3.4 Miscellaneous

This section provides options to adjust the visual appearance of the labware by giving it a (different) color or define a mesh or texture file and to define vendor, part numbers and links to websites.

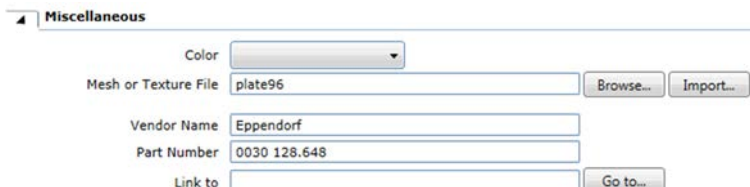


Fig. 77: Miscellaneous

8.3.5 Allowed Labware Types

In this section the labware that can be placed on top of the selected labware can be defined. For plates for example other plates can be placed in order to stack or alternatively lids can be placed onto plates.

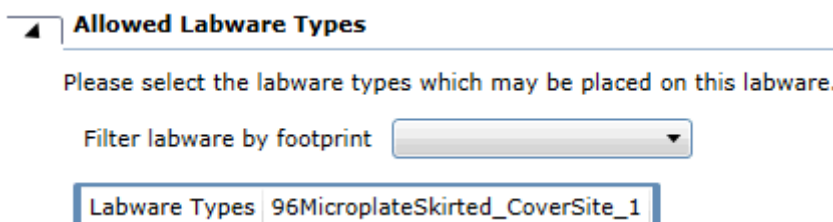


Fig. 78: Filter labware by footprint

You can specify the footprint of the labware.

Filter labware by footprint: Click the **Filter labware by footprint** arrow to see more options. The list contains the different footprints of all labware types present on the system by selecting an option in the dropdown a table appears showing all selected labware that may be placed onto the selected CoverSite of the labware.

8.3.6 Allowed Locations

This section contains information on which underlying location the labware is allowed to be placed. These are mostly carrier sites but also labware sites. A filter allows to filter for all selected, all unselected or all options.

Allowed locations

The **Allowed locations** list shows on which location the carrier or labware is allowed. Each entry comes with a check box, which you can clear or select. To narrow down the list, use **Filter locations**.

8.3.7 Robot Specific

This section is relevant if an RGA or a channel gripper is present on the system. Here gripping parameters are defined (only applicable to RGA), offsets between plates when they are stacked and whether the labware allows lids or not.

Robot Specific

Grip Narrow [mm] Wide [mm]

Stacked Plate to Plate offset X Y Z [mm]

Labware can have lid ☐

Select a Lid

Plate to Lid offset X Y Z [mm]

Fig. 79: Robot specific

8.3.8 Custom Attributes

Custom attributes are needed to define some specific robotic actions. Not all available custom attributes are needed for all labware, usually only a handful. To add specific attributes press the + button on the bottom of the table. A free text field allows to enter a custom attribute that is known or alternatively a dropdown menu allows to choose one. To delete a custom attribute press the x-button at the end of each entry. The following custom attributes are available: **Custom Attributes** are presented as a table with the following columns: **Attribute Name**, **Data Type**, and **Attribute Value**.

- **KeepForNewRun**: Indicates that the labware or tool instances of the last runtime workspace shall be taken when starting a new run
- **RenderVisualChildren**: Define if children objects of a worktable object shall be rendered
- **Static_Labware**: Define the naming rule for the labware generator
- **WasherPlateType**: Washer plate type
- **WorkspaceDeltaLevel**: Preferred workspace delta level for this labware
- **CllidSensitivityCalculationLabware**: Defines the type used for the cllid sensitivity calculation
- **InitializationOffset**: The offset which exists during initialization between an axis with no mounted tool and mounted tool
- **IsFcaLiquidWaste**: Definition of the FCA liquid wastes, as fallback the plastic waste should be used
- **MountOffset**: The offset which exists if a tip is mounted and reduces the z-dimension of the tip
- **TipCapacity**: Capacity of a single tip. Is used for context check in script commands when the user defines a volume to be aspirated
- **TipDimension**: The size of the tip in X, Y, Z
- **ToolIdName**: Identifier for labware/tool
- **ToolType**: Type identifier for labware/tool; together with ToolIdName and ToolType there shall be a unique identifier
- **Grasp_location**: Position where the labware should be gripped. This is an offset from the default grasp position
- **Grasp_location_N**: Position where the labware should be gripped when gripping narrow. Offset from the taught position for gripping the plate narrow

- **Grasp_location_W**: Position where the labware should be gripped when gripping wide. Offset from the taught position for gripping the plate wide
- **Grip_Delta**: Distance added to the grip distance when releasing the labware or lid. This offset is applied when gripping with the RGA.
- **Lid_arrival**: Movement of the robotic arm to remove a lid of a labware
- **Lid_Force**: Force to grip lid
- **Lid_grasp_location_N**: Offset from the taught position for gripping the lid of a plate in portrait orientation
- **Lid_grasp_location_W**: Offset from the taught position for gripping the lid of a plate in landscape orientation
- **Speed**: Maximum speed of robotic arm in percentage of full speed
- **SpeedX**: The movement speed of the arm used for capacitive scanning of the teaching triangles in X-direction.
- **SpeedY**: The movement speed of the arm used for capacitive scanning of the teaching triangles in Y-direction.
- **AccelerationX**: The movement acceleration of the arm used for capacitive scanning of the teaching triangles in X-direction.
- **AccelerationY**: The movement acceleration of the arm used for capacitive scanning of the teaching triangles in Y-direction.
- **DisplayToolName**: User-friendly name for Infopad, Controlbar and dropdown lists
- **DiTi_type_group**: Group of DiTi type
- **DiTiToolType**: Identifier for the DiTi type
- **GripperCenterOffset**: Offset from the rotation center to plate grip position
- **GripperSpreadOffset**: Measured between the grippers
- **GripperZOffset**: Z offset from the rotation center to plate grip position
- **LeftGripperBoundingBox1Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox1Offset**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox2Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox2Offset**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox3Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox3Offset**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox4Dimension**: Definition of bounding box for the gripper
- **LeftGripperBoundingBox4Offset**: Definition of bounding box for the gripper
- **ListNumberInDropDownMenu**: Sorting rule in steps of ten for the list of value in the script command parameter "Adapterplate"
- **RightGripperBoundingBox1Dimension**: Definition of bounding box for the gripper
- **RightGripperBoundingBox1Offset**: Definition of bounding box for the gripper
- **RightGripperBoundingBox2Dimension**: Definition of bounding box for the gripper

- **RightGripperBoundingBox2Offset:** Definition of bounding box for the gripper
- **RightGripperBoundingBox3Dimension:** Definition of bounding box for the gripper
- **RightGripperBoundingBox3Offset:** Definition of bounding box for the gripper
- **RightGripperBoundingBox4Dimension:** Definition of bounding box for the gripper
- **RightGripperBoundingBox4Offset:** Definition of bounding box for the gripper
- **MapToolName:** The Tool name used by MAP.
- **XcaCenterPositionOfFirstDiTiPositionZ:** The center position of the first tip in Z.
- **GrippingZOffset:** The offset in Z when the FCA Channel Grippers attempt to grip labware.
- **SolidWasteConsumption:** Defines how many units a tip should add to the Solid Waste Consumption counter.
- **DiTiTypeGroup:** List of tip Types which a tip waste supports.
- **TipPositionMoveIn:** Defines the first position of the tip when moving starts.
- **DropBushingPositionOther:** Defines the position of the bushing before moving starts.
- **DropBushingPositionStart:** Defines the position of the bushing when ejection starts.
- **DropBushingPositionEnd:** Defines the position of the bushing when ejection ends.
- **TipPositionEjectionMove:** Defines the position of the tip when ejection ends.
- **TipPositionMoveIn_TecanDiTi5000ul:** Defines the first position of the tip when moving starts for 5000ul tips.
- **DropBushingPositionOther_TecanDiTi5000ul:** Defines the position of the bushing before moving starts for 5000ul tips.
- **DropBushingPositionStart_TecanDiTi5000ul:** Defines the position of the bushing when ejection starts for 5000ul tips.
- **DropBushingPositionEnd_TecanDiTi5000ul:** Defines the position of the bushing when ejection ends for 5000ul tips.
- **TipPositionEjectionMove_TecanDiTi5000ul:** Defines the position of the tip when ejection ends for 5000ul tips.
- **SolidWasteCapacity:** The Capacity of the tip waste before it needs to be emptied.
- **SolidWasteUsage:** The current count of tip usage in the tip waste.
- **BoundingBoxRelativeToToolPositionFirstCorner:** The left, front, lower corner of the bounding box relative to the tool position.
- **BoundingBoxRelativeToToolPositionSecondCorner:** The upper, right, back corner of the bounding box relative to the tool position.
- **XcaSizeHook:** The size of the hooks at the end of the channel gripper. This is the part that is under the labware during transport.
- **ToolOffsetY:** The distance between the Y tool position and the drive Y position.
- **PickupZOffset:** The additional Z Pick offset for gripper pickup.

- **LiquidWasteCapacity:** The Capacity of the liquid waste before it needs to be emptied.
- **DistanceFromLabwareEdgeForPresenceCheck:** The distance from the edge of the labware in [mm] where the labware presence check is executed when using the FCA Channel Grippers.
- **FCA_Grip_Delta:** Distance added to the grip distance when releasing the labware or lid. This offset is applied when gripping with the channel grippers on the FCA.

8.4 Script Editor

The script editor is where protocols are being developed. The single script lines are also called commands and those are categorized based on arm and device within the Controlbar. The Script editor allows to develop new scripts from scratch, open existing scripts and adjust by changing order of the statements, adding new commands or deleting commands. Furthermore it allows to test and troubleshoot protocols by providing different tools ([“refer to chapter 8.4.2” \[▶ 142\]](#)).



Do not use breakpoints in validated scripts. They are intended for development purposes only.

When the **Script** editor is open (you are viewing a script), the **Commands** section of the **Controlbar** only shows script commands.

Check for correct execution:

- ✓ Wrong Application data loaded or saved or deletion not working can cause wrong results.
1. Check the execution of the correct scripts before releasing data.
 2. Check for correct execution data being loaded at end of run.

8.4.1 Variables

Also see about this

 ["Variables" ▶ 376](#)

8.4.1.1 Declaring and using Variables

vControl requires explicit declaration of all variables in the Variables section, at the bottom of the editor. The Variables section is available in the Script, Process and Method editors. New variables can either be directly added and defined in the variable pane or within the commands that use the variables (e.g. Set Variable) by using the "Declare"-button. The **declare** button will make a new entry in the variable pane, which may need to be adjusted depending on type, scope, or default value.

vControl has 3 types of variables: string (letters and numbers), floating point (numbers with decimals) and integers (numbers without decimals). Setting the type of variable will change how the input screens appear on the touch monitor. String = Full Keyboard, Floating Point = Number pad with decimal point, Integer = Number pad with no decimal point.

Name	Type	Value	Scope	Persist	Read only	Query at startup	Query text
Var1	String	abc	Parameter	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Var2	String	def	Iteration	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Var3	Integer	0	Run	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Var4	Floating Point	1.1	Script	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Fig. 80: Variables

Variables in vControl can have one of five scopes (**Run**, **Script**, **Iteration**, **Parameter**, and **Process**).

For more information on the use of variables by the vControl **Scheduler**, see section “[Variables](#)” [▶ 376].

Run: Variables with this scope are shared globally by all the scripts in a run.

Script: Variables with this scope are declared in scripts with the **Set Variable** command. They are not shared with other scripts. If you declare a variable with scope **Run** or **Iteration**, do not declare it again within a script. This will cause an error.

Iteration: Iteration scope is like script scope. However, these values reset to the defaults with each new iteration at run-time. Only impacts Methods that use Scripts or with more than one iteration.

Parameter: Variables of scope **Parameter** are automatically added to the “Transfer values to subroutine” section of the Module and Subroutine commands. The behavior of variables with scope Parameter are otherwise identical to variables with scope Script.

Process: Values of a variable with scope **Process** are only valid and accessible for a specific Process during the run but are not be accessible to other Processes in the run.

Persisted Variable Values

The application allows to persist variable values of variables with scope Run. The current values of these variables are stored at the end of each run and retrieved when a Method is created or loaded. Only variable of scope Run support persistence.

For other scopes the Persist check box is disabled. This feature is especially relevant for Scheduler Cell-based Assay workflows (“[Support for Cell-Based Assays](#)” [▶ 387]). If variable values shall not be lost when a run is stopped or the software is restarted, the affected variables can be set to persist their values. To enable persistence, check the Persist checkbox in the variable definition in the Process or Script declaring the variable. If the Persist checkbox is unchecked, but a stored value is available, the user will be warned that the stored values will be deleted. If the user rejects the deletion, the check box will be checked again.



If a variable is not declared as persisted but a stored value matches the variable definition (name and type) its value will be used when a method is opened. This will be reflected in the Variables section of the Method editor by showing the Persist checkbox as checked.

8.4.2 Test and Troubleshoot a Script

vControl provides multiple features to help testing or troubleshooting a script. 1) Context Checking -> see next chapter 2) Loop iteration simulation Since vControl is object oriented, the worktable will always show the current Labware Position - both at runtime and within the script editor when clicking through the script lines. The script option Loop Counter helps outlook the current Labware position during every cycle of a given loop. The Loop counter variable parameter corresponds to the loop counter for the selected loop in the script, whereas the Target Value parameter corresponds to the loop cycle to be shown on the worktable. 3) Disabling command lines or setting breakpoints: By right-clicking on a specific command line the command can either be disabled, i.e. it will be ignored at execution or a breakpoint can be set, which will stop the execution at this command until the user confirms to continue. 4) Different run modes provided by clicking on the small arrow on the Run button:

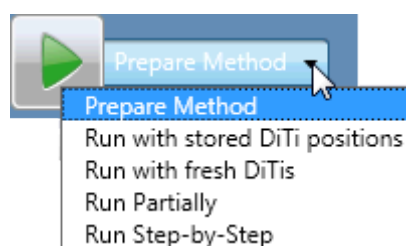


Fig. 81: Script editor

Run with stored DiTi positions: Run with stored DiTi positions resets all labware positions on the worktable but preserves disposable tip information for from last run, including the positions of tip trays, boxes, and layers in nested tip stacks. This applies to Flexible Channel Arm and Multiple Channel Arm tip types.

Run with fresh DiTis: Run with fresh DiTis resets all labware positions on the worktable and stored disposable tip information from the last run.

Run Partially: Run Partially the current selected lines of the script.

Run Step-by-Step mode: Before the start of each command line in the script, the user must click continue in the runtime controller or the touch monitor.



Subroutines: If a synchronous subroutine is present in the script, the complete subroutine runs in step-by-step mode as well. If an asynchronous subroutine is present in the script, then, before the start of this command the user must click continue, but the asynchronous subroutine itself is not run in step-by-step mode.

It is expected that once a script is developed it will be prepared as a method and the operator will execute this method via the interface on the touch monitor.

In addition to these four run modes, vControl offers the ability to deactivate script lines or set breakpoints at script lines by right clicking on the command. Deactivated script lines and breakpoints will also be ignored in sub-routines.

Script run is not allowed to start if open editors have unsaved changes. A dialog box will ask whether to save unsaved changes. Press **yes** on the dialog box to save unsaved changes in all open editors and continue with run.

8.4.3 Context and Syntax Check

The context check feature validates each script line and its parametrization in the context of other commands, the worktable and liquid class entries. It checks whether certain positions can be reached, whether pipetting actions can be performed with given labware, volume and liquid class or whether a tip can be picked up. Syntax checking is very similar but it checks whether the values entered in a command are in a certain format and are valid. Both features are automatically enabled and will raise an error during edit time as form of red indicator and some description in the infopad of the Controlbar. If by any means this feature shall be disabled this is possible via the Run menu -> disable "Auto Context check". All previous errors are now still marked but only as warnings in yellow, such that the script can be executed. The user takes now responsibility for certain potential crashes.

8.5 Method Editor

Use the Method editor to create executable protocols consisting of one or more scripts and/or processes with one or more iterations. The Method editor consists of 3 main sections: - Main panel to define scripts and processes to be used in the method - Lower panel with different sections called "Method properties" to define main properties of the method "Variables" and "Touch Tools Settings" - Worktable window Important: The worktable needs to be consistent across several scripts and processes used within the method.

You can drag **Scripts** to your method.

1. Choose **Controlbar > Category > Scripts**.
2. Dropping the element in place.
3. Move the script to the place shown in the following figure.

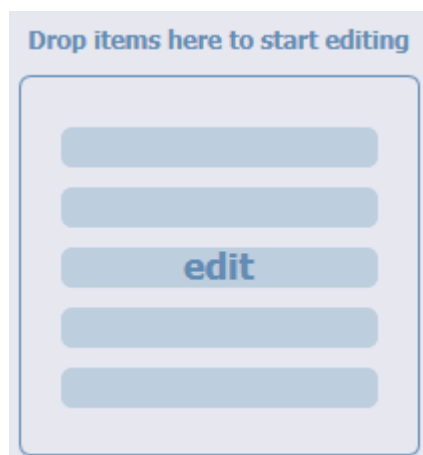


Fig. 82: Drag-and-drop feature: The edit zone

The next figure shows the editor after dropping the element in place.

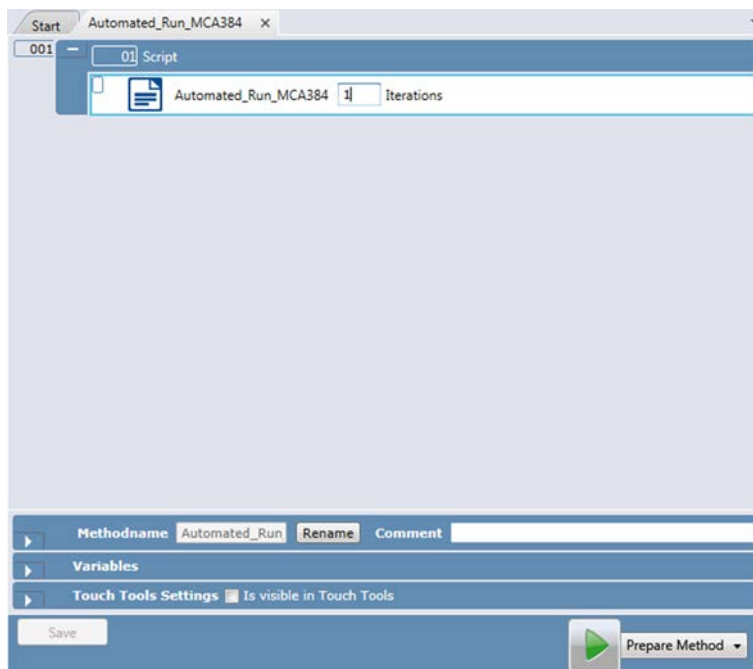


Fig. 83: Method editor with a dropped Script item



Although the edit zone is no longer visible but replaced by the dropped item, the drag-and-drop feature is still available. Think of it as an invisible drop zone in the background.

Iterations: Enter the number of iterations a Script or Process is being executed.

8.5.1 Method Properties

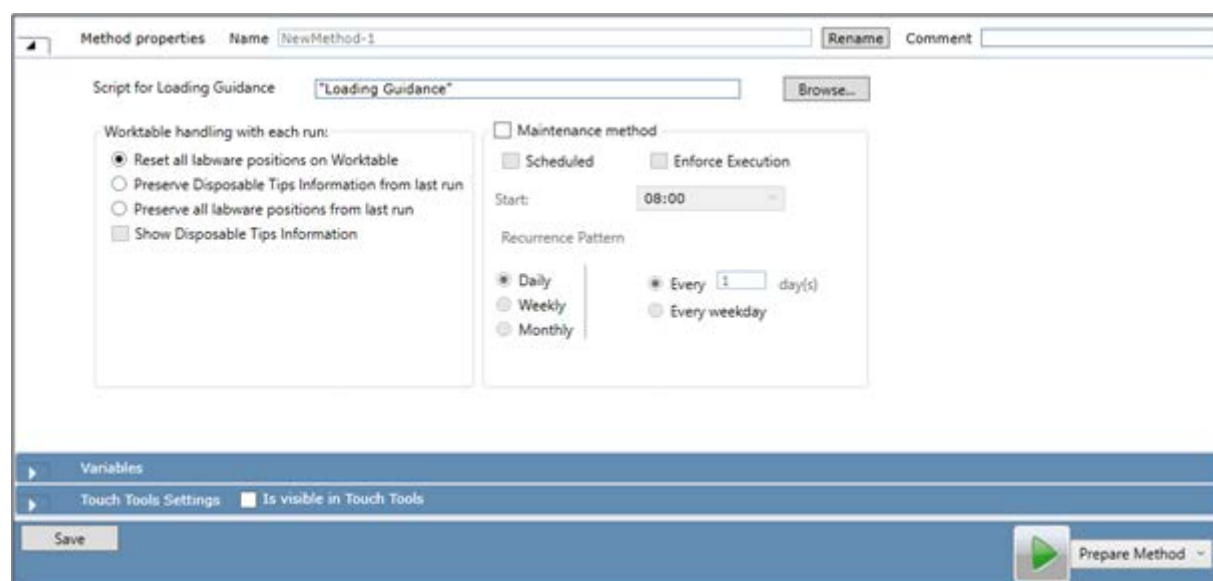


Fig. 84: Method Editor

Method Name: Rename the method name.

Comment: Add a Comment.

Script for Loading Guidance: It is possible to select a script to guide the operator through loading an instrument. This script is always executed at the beginning of the method. The instrument does not have to be initialized to execute the Loading Guidance. The Door can be open during the execution of this script. Only command which do not move the configured arms of the instrument are allowed in the defined **Script for Loading Guidance**. If a script is incompatible, a context check error message will be shown.

Worktable Handling with each run:

Reset all labware positions on Worktable: This option re-sets all stored tip position from last run and a new run will start picking up tips from position 1 for all tip boxes and from the positions defined on the worktable.

Preserve Disposable Tips Information from last run: This option keeps the Tip positions and tip information from the last run if the same worktable is used. However, changes made in the same worktable for tip (box) positions together with this option selected is not effective until a run has started with the option **Reset all labware positions on Worktable** since the stored information from last run is always used.

Preserve all labware positions from last run: This option keeps the position of all labware like tips, plates, troughs etc. from last run if the same worktable is used. However, changes made in the same worktable for labware positions together with this option selected is not effective until a run has started with the option **Reset all labware positions on Worktable** since the stored information from last run is always used.

Is Visible in TouchTools: You can look at the variables and edit the Touch Tools Settings. Select **Is visible in TouchTools** to make methods appear to the operator on the touch monitor.

Maintenance method check box: A method will show up on the touch monitor under system care if you select the **Maintenance method** check box and additionally the **Is visible in TouchTools** checkbox in the **TouchTools Settings** section below. It is possible to define a Schedule for this maintenance method by selecting the **Scheduled** checkbox. The **Start** field will define a time when the maintenance method shall be executed. The **Recurrence Pattern** defines the execution interval of the maintenance method. The checkbox **Enforce Execution** will force the user to execute the maintenance methods before executing any other methods or scripts.

8.5.2 Touch Tools Settings

The **Touch Tools Settings** check box allows the method to be visible on the touch monitor.

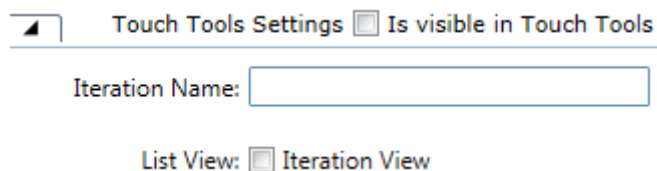


Fig. 85: Is visible in Touch Tools

Iteration Name: Optionally a name can be defined for the Iterations, which will then be visible in the TouchTools.

Is visible in Touch Tools: Select **Is visible in Touch Tools** to make methods appear to the operator on the touch monitor.

List View: To add the iteration controls to the method on the touch monitor when Prepare Method, select the **Iteration View** checkbox.

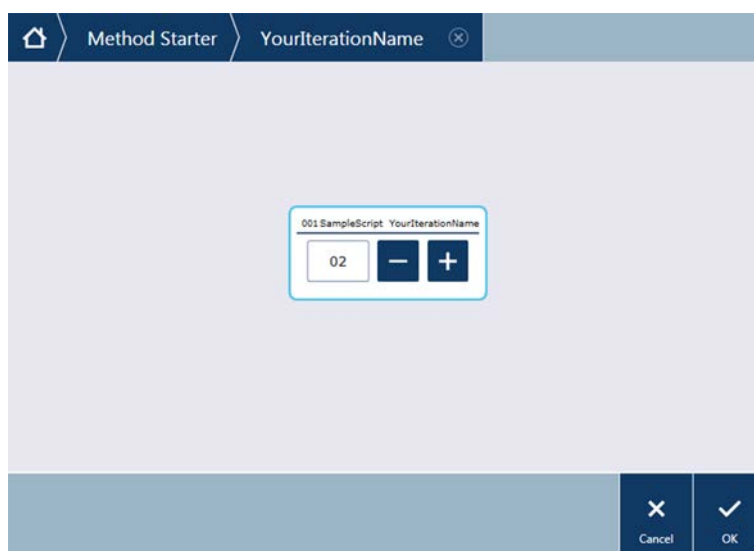
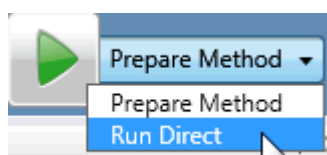
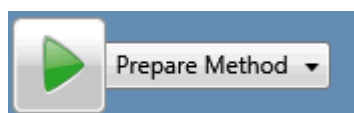


Fig. 86: Touch Tools: Example of a method with Iteration View

8.5.3 Prepare Method or Run Direct



Method run is not allowed to start if open editors have unsaved changes. A dialog box will ask whether to save unsaved changes. Press **yes** on the dialog box to save unsaved changes in all open editors and continue with run.

8.6 Liquid Class Editor

A liquid class defines the robotic actions that will take place during a pipetting step. It includes robotic movements into and out of the well, Positioning parameters, detection parameters, liquid handling -related parameters like speed, airgap volumes, correction volumes or dispense modes. .

Several liquid class templates exist for different pipetting sequences (single versus multi, with or without mixing or pure mixing). Furthermore liquid classes are usually defined for a specific liquid, because liquid properties like liquid conductivity or viscosity may play an important role on the needed pipetting parameters. However, usually liquid classes are not specific for certain tip types or volumes, as most parameters include tip-type or volume-dependent settings. Special liquid classes also exist for e.g. only detecting a liquid but not for pipetting something.

The liquid class Editor is structured into several sections, where each liquid class template contains of different sections depending on the pipetting sequence. Common to all different liquid classes is the "General" section, the "Advanced" section, the name and comment field and the Save buttons. The following chapters will describe in more detail the different sections and to which liquid classes they apply.

8.6.1 General



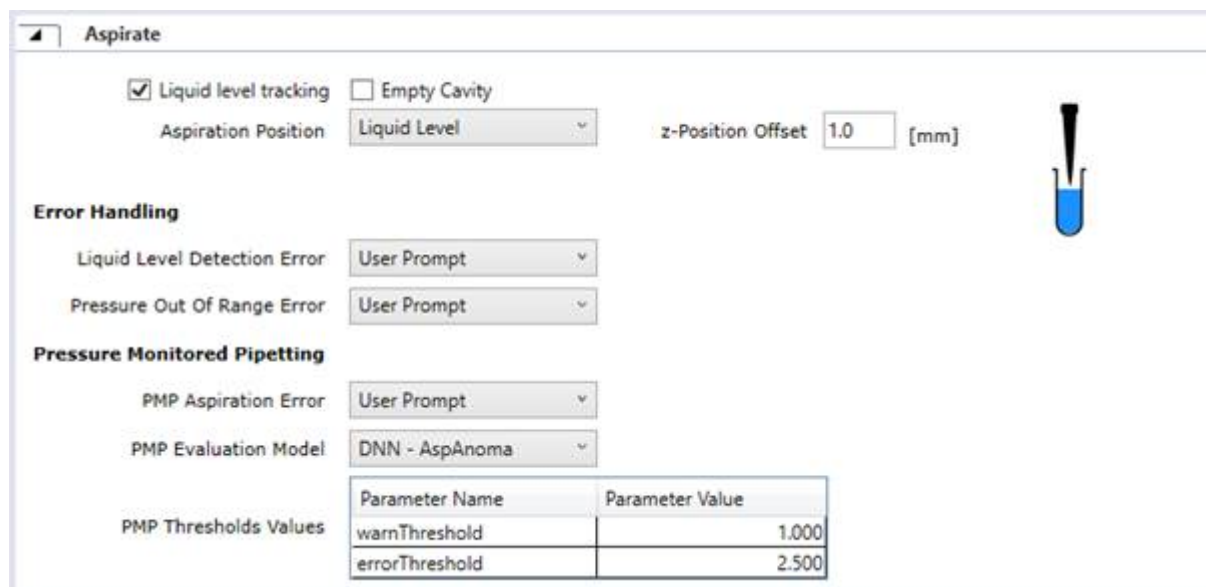
The general section contains parameters that are global and independent on the liquid class type. The picture gives an indication of the way the liquid class works.

Liquid Name: Defines the name of the liquid with all associated liquid properties. Changes done here impact the pipetting behaviour.

Liquid Level Detection Mode: Defines how the liquid level is detected. This can be via capacitive -based detection (cLLD), pressure-based detection (pLLD) or no liquid level detection.

Pressure Detection Mode: If pressure-based detection is selected a further dropdown menu to select the type of pressure detection mode. By default **Aspirate Detection** is selected, which works for most liquids and hence it's not recommended to change this

8.6.2 Aspirate



Aspirate

☒ Liquid level tracking ☐ Empty Cavity

Aspiration Position: Liquid Level

z-Position Offset: 1.0 [mm]

Error Handling

Liquid Level Detection Error: User Prompt

Pressure Out Of Range Error: User Prompt

Pressure Monitored Pipetting

PMP Aspiration Error: User Prompt

PMP Evaluation Model: DNN - AspAnoma

PMP Thresholds Values

Parameter Name	Parameter Value
warnThreshold	1.000
errorThreshold	2.500

The Aspirate section contains parameters and error handling options related to aspirate actions. This section is equal in all liquid classes.

Liquid level tracking: Defines whether the tip shall move down with the liquid level during aspiration.

Empty Cavity: In case this option is checked, the liquid class will overaspirate liquid from the well ignoring available liquid, therefore not raising an error. Note: This is not recommended for general pipetting workflows, where sample tracking needs to be correct.

Aspiration Position: Defines the starting or reference position of the aspirate step.

z-Position Offset: Defines an offset in positive (down) or negative (up) direction relative to the **Aspiration Position**. Note: The image is dynamic and will show approximate positioning depending on the selection.

Error Handling

Liquid Level Detection Error: Defines the automatic error handling in case an error like “Not enough liquid” occurs during aspiration. The options are “Aspirate anyway”, “Aspirate available volume”, “Deactivate Tip” (which includes pipetting actions until the tip is dropped) and “User Prompt”. During a user prompt there are additional error handling options “Refill Labware”, “Retry” or “Abort Run”.

Pressure Out of Range Error: Defines the automatic error handling in case of an “Pressure profile error detection”, which means that the pressure curve detected some anomaly resulting from e.g. a clot or foam aspiration. The options are “Continue, Pipette Nothing”, “Continue, Dispense Back, Pipette Nothing”, “Continue, Drop Tip, Pipette Nothing”, “Retry once with new tip” or “User Prompt”. During a user prompt the same error handling options will be available.

Pressure Monitored Pipetting (PMP AI)

This section is only available when a licence is present for that feature or in a simulated environment.

PMP Aspiration Error: Defines the automatic error handling in case the PMP AI module detected pipetting anomalies resulting from e.g. a clot, foam or bubble aspiration. The options are “Ignore and Continue”, “Retry once with new tip”,

“Dispense back to source and deactivate tip”, “Dispense to waste and deactivate tip” or “User Prompt”. During a user prompt the same error handling options will be available.

PMP Evaluation Model: Defines the model that is used to evaluate the pipetting. There is currently one model available that is trained on Whole Blood and is an AnomalyDetection deep neural net model for aspiration sequence. These parameters need to be fine-tuned depending on the pipetting use case.

PMP Thresholds Values: Only available for “AnomalyDetection”-based models. Depending on the selected model the default values for warnThreshold and errorThreshold can be different

8.6.3 Dispense

Dispense			
Dispense Mode	ContactEdge	Z-Position Offset	-1.0 [mm]
Empty Tips Dispense Mode	FreeDispense	Edge Offset in X	0.0 [mm]
Liquid Level Detection Error	User Prompt	Edge Offset in Y	0.0 [mm]

The Dispense section contains parameters and error handling options related to dispense actions. This section is equal in all liquid classes.

Dispense Mode: Defines the way the liquid is dispensed. The options are “Free Dispense” (dispense in air not touching the liquid), “Contact Wet” (dispense at liquid level touching the liquid), “Contact Dry” (dispense on the bottom of a dry well), “Contact Edge” (dispense at the edge of a well above liquid level). Note: For Contact Edge the appropriate offsets for the labware in X and Y need to be defined. The offsets can be overwritten by defining X and Y offsets in the pipetting script commands.

Empty Tips Dispense Mode: Defines the way the liquid is dispensed when performing an empty tip command. The options are the same as for Dispense Mode “Free Dispense” (dispense in air not touching the liquid), “Contact Wet” (dispense at liquid level touching the liquid), “Contact Dry” (dispense on the bottom of a dry well), “Contact Edge” (dispense at the edge of a well above liquid level).

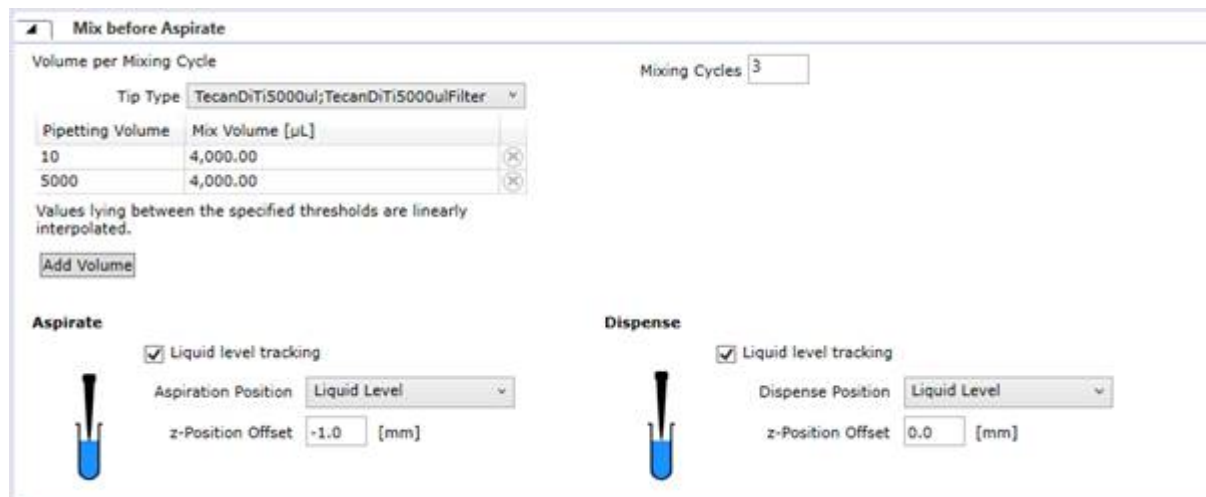
z-Position Offset: Defines an offset in positive (down) or negative (up) direction relative to the Dispense Position. Not applicable for Contact Edge Dispense Mode. Note: The image is dynamic and will show approximate positioning depending on the selection.

Edge Offset in X: Defines the offset in X direction from the center of the cavity for the Contact Edge Dispense mode.

Edge Offset in Y: Defines the offset in Y direction from the center of the cavity for the Contact Edge Dispense mode.

Liquid Level Detection Error: Defines the automatic error handling in case an error like “Liquid level not detected” occurs during dispense action. The options are “Deactivate Tip”, “Ignore and Continue” and “User Prompt”. During a user prompt there are additional error handling options “Retry” or “Abort Run”.

8.6.4 Mix before Aspirate or Mix after Dispense



This section only exists in liquid classes that can perform a mix before Aspirate or a mix after Dispense. The section is divided into the upper section that applies to both Aspirate and Dispense action within mixing and the lower part contains a section each for both Aspirate and Dispense. Note: The error handling options are the same as defined in the Aspirate and Dispense sections respectively.

Volume per Mixing Cycle: Defines the volume to be aspirated and dispensed in each mixing cycle. This parameter can be defined for each tip type and in dependence of the main pipetting volume for the Aspirate-Dispense step. The table needs to contain at least 2 datapoints and can be extended with as many data points as needed to fine-tune mixing volume for a range of individual pipetting volumes. Values in between datapoints are linearly interpolated and rounded to full integers. Values outside of the defined datapoints get the same value as the lowest or highest defined volume respectively. For example the pipetting volume 10ul is associated with a mixing volume of 8ul and the second datapoint 100ul has a mixing volume of 80ul, therefore a pipetting volume of 50ul contains a mixing volume of 40ul. Data doesn't get extrapolated outside of defined datapoints.

Mixing Cycles: Defines the number of mixing times.

Aspirate

Liquid level tracking: Defines whether the tip shall move down with the liquid level during aspiration.

Aspiration Position: Defines the starting or reference position of the aspirate step.

z-Position Offset: Defines an offset in positive (down) or negative (up) direction relative to the Aspiration Position. Note: The image is dynamic and will show approximate positioning depending on the selection.

Dispense

Liquid level tracking: Defines whether the tip shall move up with the liquid level during dispense.

Dispense Position: Defines the starting or reference position of the dispense step.

z-Position Offset: Defines an offset in positive (down) or negative (up) direction relative to the Dispense Position. Note: The image is dynamic and will show approximate positioning depending on the selection.

8.6.5 Mixing Only

The screenshot shows the 'Mixing' configuration window. It is divided into two main sections: 'Aspirate' and 'Dispense'. Each section features a pipette icon and a set of controls. In the 'Aspirate' section, 'Liquid level tracking' is checked, 'Aspiration Position' is set to 'Liquid Level', 'z-Position Offset' is '-1.0 [mm]', and three error handling options are set to 'User Prompt'. In the 'Dispense' section, 'Liquid level tracking' is also checked, 'Dispense Position' is set to 'Liquid Level', 'z-Position Offset' is '0.0 [mm]', and the 'Liquid Level Detection Error' is set to 'User Prompt'.

For pure mixing liquid classes there is no Aspirate or Dispense section, only a Mixing section.

Aspirate

Liquid level tracking: Defines whether the tip shall move down with the liquid level during aspiration.

Aspiration Position: Defines the starting or reference position of the aspirate step.

z-Position Offset: Defines an offset in positive (down) or negative (up) direction relative to the **Aspiration Position**. Note: The image is dynamic and will show approximate positioning depending on the selection.

Liquid Level Detection Error: Defines the automatic error handling in case an error like "Not enough liquid" occurs during aspiration. The options are "Aspirate anyway", "Aspirate available volume", "Deactivate Tip" (which includes pipetting actions until the tip is dropped) and "User Prompt". During a user prompt there are additional error handling options "Refill Labware", "Retry" or "Abort Run".

Pressure Out of Range Error: Defines the automatic error handling in case of an "Pressure profile error detection", which means that the pressure curve detected some anomaly resulting from e.g. a clot or foam aspiration. The options are "Continue, Pipette Nothing", "Continue, Dispense Back, Pipette Nothing", "Continue, Drop Tip, Pipette Nothing", "Retry once with new tip" or "User Prompt". During a user prompt the same error handling options will be available.

Dispense

Liquid level tracking: Defines whether the tip shall move up with the liquid level during dispense.

Dispense Position: Defines the starting or reference position of the dispense step.

z-Position Offset: Defines an offset in positive (down) or negative (up) direction relative to the **Dispense Position**. Note: The image is dynamic and will show approximate positioning depending on the selection.

Liquid Level Detection Error: Defines the automatic error handling in case an error like "Liquid level not detected" occurs during dispense action. The options are "Deactivate Tip", "Ignore and Continue" and "User Prompt". During a user prompt there are additional error handling options "Retry" or "Abort Run".

8.6.6 Multi Pipetting

Multi Pipetting

☒ Retake Trailing Air Gap after each Dispense

First Dispense Correction

Conditioning Volume: ConstantVolume

Last Dispense Correction

Excess Volumes

Tip Type: TecanDti5000ul;TecanDti5000ulFilter

Aliquot Volume	Excess Volume [µL]
10	0.00
5000	0.00

Values lying between the specified thresholds are linearly interpolated.

Add Volume

Dependent Volumes

Tip Type: TecanDti5000ul;TecanDti5000ulFilter

Aliquot Volume	Conditioning Volume [µL]
10	0.00
5000	0.00

Values lying between the specified thresholds are linearly interpolated.

Add Volume

This section only applies to Multi-pipetting liquid classes.

Retake Trailing Air Gap after each Dispense: This option is activated by default, so that an air gap is aspirated automatically after each aliquot is dispensed.

First Dispense Correction

In a multi-pipetting sequence the conditions for the first aliquot differ from the conditions of the other aliquots and therefore often the first aliquot needs to be corrected. This is achieved by introducing a Conditioning volume that is aspirated and then again dispensed automatically before the actual pipetting volume gets aspirated. This conditions the tip in a way to make the first aliquot resemble all the other aliquots.

Conditioning Volume: The parameter can either be defined as a “Nominal Volume”, which then equals the size of one aliquot or by a “Constant Volume”, which is constant independent of the aliquot volume. Alternatively, the conditioning volume can be disabled by selecting the option “Disabled”.

Dependent Volumes: This table gets active when “Constant Volume” for **Conditioning Volume** is selected. A constant volume can be defined for each tip type and in dependence of the aliquot volume. The table needs to contain at least 2 datapoints and can be extended with as many data points as needed to fine-tune the constant conditioning volume for a range of individual pipetting volumes. Values in between datapoints are linearly interpolated and rounded to full integers. Values outside of the defined datapoints get the same value as the lowest or highest defined volume respectively. For example: The aliquot volume 20ul is associated with a conditioning volume of 10ul and another aliquot volume of 100ul has a conditioning volume of 50ul, therefore an aliquot volume of 60ul would get a conditioning volume of 30ul. Data will also get extrapolated outside of the defined datapoints.

Last Dispense Correction

In a multi-pipetting sequence the conditions for the last aliquot differ from the conditions of the other aliquots and therefore often the last aliquot needs to be corrected. This is achieved by introducing an **Excess volume** that is aspirated before the actual pipetting volume gets aspirated. This makes the last aliquot resemble all the other aliquots.

Excess Volume: The parameter can be defined per tip type and also in dependence of the aliquot volume.

The table needs to contain at least 2 datapoints and can be extended with as many

data points as needed to fine-tune the excess volume for a range of individual pipetting volumes. The value in between two datapoints will get the same value as the lower datapoint within the range. Values outside of the defined datapoints get the values from the lowest or the highest datapoint respectively.

8.6.7 Advanced Section

Advanced

cLLD Sensitivity

Aspiration (Least Sensitive) 0..1 (Most Sensitive)

Dispense (Least Sensitive) 0..1 (Most Sensitive)

Mix (Least Sensitive) 0..1 (Most Sensitive)

pLLD Parameters

Average Samples ⓘ

Detection Time [100µs] ⓘ

Detection Threshold [0.1 mBar/digit] ⓘ

Speed

LLD Speed [mm/s]

Retract Speed [mm/s]

☒ Stop Pipetting on Pressure Profile Error Detection

Tank Pressure

Tip Types	Negative [mbar]	Positive [mbar]
TecanDiTi5000ulFilter;TecanDiTi5000ul	750.00	1,750.00
TecanDiTi1000ulWbFilter;TecanDiTi1000ulWb	750.00	1,750.00
TecanDiTi1000ulFilter;TecanDiTi1000ul	750.00	750.00
TecanDiTi200ulFilter;TecanDiTi200ul	750.00	750.00
TecanDiTi50ulFilter;TecanDiTi50ul	750.00	750.00
TecanDiTi10ulFilter;TecanDiTi10ul	750.00	750.00

The higher the absolute pressure values, the higher the pipetting speeds.

Volume

Trailing Airgap (TAG)

Tip Type

Aspirate Volume	Trailing Air Gap [µL]
10	20.00
5000	20.00

Values lying between the specified thresholds are linearly interpolated.

Blowout Volumes

Tip Type

Aliquot Volume	Blowout Volume[µL]
10	50.00
5000	50.00

Values lying between the specified thresholds are linearly interpolated.

Delay

After Aspirate

Tip Type

Pipetting Volume [µl]	Delay [ms]
10	200.00
100	200.00
1000	200.00
5000	200.00

Values lying between the specified thresholds are linearly interpolated.

After Dispense [ms]

Correction Volumes

Tip Type

Aliquot Volume	Correction Volume[µL]
0	0.00
4800	0.00

Values lying between the specified thresholds are linearly interpolated.

pLLD Parameters: Those parameters change sensitivity of pLLD detection. Lower Average Sample size, Detection Time and Detection Threshold increases the sensitivity. Higher Average Sample size, Detection Time and Detection Threshold decreases the sensitivity. Usually these values don't have to be changed.

LLD Sensitivity: This parameter is to adjust sensitivity values for liquid level detection for all actions Aspirate, Dispense and Mix individually. Usually these values don't have to be changed and it's also not recommended to change these values.

LLD Speed: Defines at which z-speed the tip moves when performing liquid level detection. The higher the speed the more inaccurate the detected height of the liquid level. **Retract Speed:** Defines the z-speed the channel moves out of the well after pipetting.

Tank Pressure: The tank pressures inside the negative and positive pressure tanks respectively, have an impact on liquid flow speed in the tip. The higher the absolute pressure in either negative or positive pressure tank, the higher the liquid flow speed in the tip during aspirate or dispense, respectively for a given tip type. If different tip types are used within a workflow it is recommended to keep the pressures for different tip types as similar as possible in order to reduce waiting time before pipetting steps due to preparation of the required target pressure in the pipetting system. Therefore, the default values for different tip types are equal, except for the 5000ul tip.



There is a limitation of possible pressure values for maximum values (not minimum values). For the positive pressure tank this is 1750mbar overpressure, for the negative pressure tank it's 750mbar underpressure or 220mbar absolute pressure. Higher absolute values will be reduced to the threshold values and a warning will be logged.

Stop Pipetting on Pressure Profile Error Detection: This option is activated by default for non-mixing liquid classes. It defines the behaviour of the system after a Pressure Profile Error, which indicates a potential clot or foam aspiration has been detected. If the option is activated the pipetting stops immediately and depending on what was configured in the automatic error handling option the software reacts accordingly, allowing the user to recover from the error. After such an event has been triggered, the software performs a channel-self check automatically after the tip has been dropped in order to verify whether the inline filter got wet. If that's the case, the tip automatically gets deactivated and can't be further used until the inline filter gets replaced or another channel-self check has passed. If this option is turned off, a manual Channel self-check is recommended. For this use the Script command "Channel Self-Test" in your script.

Trailing Airgap Volume (TAG): Defines the airgap that is taken up after the last aliquot in order to prevent liquid dropping out of the tip. The parameter can be defined per tip type and also in dependence of the aspirate volume. The table needs to contain at least 2 datapoints and can be extended with as many data points as needed to fine-tune the TAG volume for a range of individual pipetting volumes. The value in between two datapoints will get the same value as the lower datapoint within the range. Values outside of the defined datapoints get the values from the lowest or the highest datapoint respectively.

Blowout Volume: Defines the air volume that gets dispensed after the last aliquot has been dispensed making sure to fully empty the tip. The parameter can be defined per tip type and also in dependence of the aliquot volume. The table needs to contain at least 2 datapoints and can be extended with as many data points as needed to fine-tune the blowout volume for a range of individual pipetting volumes. The value in between two datapoints will get the same value as the lower datapoint within the range. Values outside of the defined datapoints get the values from the lowest or the highest datapoint respectively.

Correction Volume: Defines a correction volume for the pipetting volume of interest to adjust accuracy of the aliquot. The parameter can be defined per tip type and also in dependence of the aliquot volume.

The table needs to contain at least 2 datapoints and can be extended with as many data points as needed to fine-tune the correction volume for a range of individual pipetting volumes. The value in between two datapoints get linearly interpolated. Values outside of the defined datapoints get the values from the lowest or the highest datapoint respectively. Note: Negative correction volumes are allowed but can lead to warnings during pipetting.

Delay after Aspirate: Defines a time delay after the aspiration step to allow the liquid to flow into the tip completely before retracting. For viscous liquids this time needs to be higher than for non-viscous liquids due to their lower flow speed. The parameter can be defined per tip type and also in dependence of the aliquot volume. This parameter also depends on the liquid chosen in the General setting of the liquid class and changes whenever the liquid type is changed.

The table needs to contain at least 2 datapoints and can be extended with as many data points as needed to fine-tune the delay after aspirate for a range of individual pipetting volumes. The value in between two datapoints get linearly interpolated. Values outside of the defined datapoints get the values from the lowest or the highest datapoint respectively.

Delay after Dispense: Defines a time delay after the dispense step to make sure the tip is emptied before retracting. For viscous liquids this time needs to be higher than for non-viscous liquids due to their lower flow speed.

9 Configure Your System

related to hardware were already done with the Setup software and don't have to be re-done. However, there are two further actions to be done. Teaching and Activation of the drivers. Furthermore there are several settings for fine-tuning the system to certain needs that can be done in the Configure System menu if it's on a global scale or in the respective editors of a carrier or labware.

vControl enables you to define an object (e.g., a microplate) once, and then to use it at any required position on the worktable. This is because vControl uses a hierarchical system for the coordinates:

- **Labware coordinates:** The labware coordinates are relative to the site on which the labware is placed.
- **Site coordinates:** The site coordinates are relative to the carrier.
- **Carrier offsets:** The carrier offsets are relative to the grid position on the worktable.
- **Grid positions:** The grid positions on the worktable have a periodic spacing of 25 mm.

This hierarchy allows you to place the labware at any suitable site, carrier and worktable position without having to redefine its coordinates.

The following lists the items you need to configure when setting up the pipetting instrument in vControl (some of the items refer to options that may not be fitted to your instrument):

- **Choose a Carrier:** When you set up the worktable for your vControl script, you first choose a carrier with the required number of sites. Each of the sites can hold labware. If you cannot find the carrier you need, you can define a new carrier by duplicating an existing one. Be aware that you are duplicating from a similar carrier. Otherwise this can lead to arms colliding with the worktable. Special carriers are hotels, carousels, stackers and passive stacks. Bounding boxes of labware or carriers placed onto those special carriers will not be considered by the PathFinder.
- **Choose the Labware:** You then choose the labware you want to use. If you cannot find the labware you need, you can define a new labware item.
- **Position the Carriers:** Position the carriers on the worktable and place the labware onto the carriers.
- **Editing an existing Script:** The procedure to follow to set up a script is described in detail in Editing an existing script, refer to section Editing an Existing Script.



The vControl database already contains a large selection of pre-defined carrier and labware types. You only need to create or edit carriers or labware if you cannot find a standard item that is suitable for your needs.

Also see about this

 ["Reference Documents" ▶ 9](#)

9.1 Teaching

The software allows to calibrate the positions of sites and devices on the worktable to compensate for mechanical variances on the instrument and therefore improve accuracy for pipetting, labware moves etc. This calibration process is called "teaching" and should be performed individually for each instrument, as each instrument will have their own installation or manufacturing related mechanical variances. The taught calibration values (also called "corrections") are then stored on the instrument (Te-Control). This way, the appropriate corrections are always used, no matter which computer is connected to the instrument.



Teaching is applied to sites, where a carrier or labware is placed on, the physical definition of a carrier remains untouched by this process.

In general, teaching or site adjustment is done in either of two ways - manual or automatic. Manual teaching is supported for all arms, automatic teaching only for the FCA. For the RGA teaching is done using the section **Robot Vectors** in the respective carrier editor ("see chapter 8.2.5" [▶ 121]), where both Site Adjustment and Vector teaching is done. For the FCA there are two different locations, depending on manual or automatic adjustment. For manual site adjustment use the section "Positions/Teaching" in the Labware Editor ("see chapter 8.3.2" [▶ 130]). Automatic teaching however is done within the Carrier Editor under section "Automatic Site Adjustment" ("see chapter 8.2.6" [▶ 125]).

Also see about this

- 📖 "Manually Mount Tool (FCA)" ▶ 175
- 📖 "Placement of Non-Grid-Based Devices" ▶ 118

9.1.1 Manual Teaching

Manual teaching is supported for all arms, automatic teaching only for the FCA. For the RGA teaching is done using the section "Robot Vectors" in the respective carrier editor ("see chapter 8.2.5" [▶ 121]), where both Site Adjustment and Vector teaching is done. For the FCA there are two different locations, depending on manual or automatic adjustment. For manual site adjustment use the section "Positions/Teaching" in the Labware Editor ("see chapter 8.3.2" [▶ 130]). Automatic teaching however is done within the Carrier Editor under section "Automatic Site Adjustment" ("see chapter 8.2.6" [▶ 125]).

Only vControl: When teaching manually, the arm gets moved by the user either using the ZeroG-mode (triggered in MoveTool) to move the arm manually or by using the MoveTool to the required position (e.g. center of a microplate well), then press "Show delta" button, which activates the "Save" button, which then stores the correction values on the instrument.



To get accurate teaching values, first mount tips with the FCA or grip a plate with the RGA using the direct commands before moving the arm to the respective site.

Also see about this

- 📖 "Manually Mount Tool (FCA)" ▶ 175
- 📖 "Placement of Non-Grid-Based Devices" ▶ 118

9.1.2 Automatic Teaching

Automatic teaching makes use of the capacitive sensing technology of the instrument. For this so called "Teaching pins" and a "Teaching Plate" are required. After mounting the pins using the direct command "Manually mount Tool" ("see [chapter 11.1.8](#)" [▶ 175]) and placing the plate to the selected position on the worktable, pressing the button "Teach" starts the procedure of automatic scanning. For this the doors need to be closed. The values are then stored on the instrument, no further user interaction is required anymore.







The calculated offsets also apply when using Channel Grippers (independent whether teaching was performed manually or automatically).



When teaching is executed on a Veya instrument with an open door, the accuracy will be reduced. This is due to the speed of the arms being reduced to 30% to comply with the safety concept of Veya. To increase the accuracy while teaching, please close the door when the arms are moving.

Also see about this

-  ["Automatic Site Adjustment" ▶ 125](#)
-  ["Positions / Teaching" ▶ 130](#)
-  ["Robot Vectors" ▶ 121](#)
-  ["Placement of Non-Grid-Based Devices" ▶ 118](#)

9.1.3 Teaching Procedure

When teaching a carrier or labware item, the following steps shall be followed:

1. When starting the teaching procedure, it is advised to have a teaching specific worktable (for example, the base worktable without any labware or extra adapters placed on it).
2. Select this teaching specific worktable before executing the teaching process.
3. Mount the teaching pin to a channel by executing the Direct Command **Manually Mount Tool**.
4. Place the teaching plate onto the site which is being taught.
5. Open the nest or carrier that should be taught in edit mode in vControl. Go to section **Automatic Site Adjustment**
6. Select the correct site and arms (if applicable).
7. Execute the teaching by pressing **Teach** followed by **Apply** to store the values onto Te-Control.

The following steps shall be followed when placing multiple devices, such as incubators, using Device Placement. Not following these steps can lead to incorrectly placed devices, or their respective Direct Commands not being shown.

- Duplicate the driver and configure it as needed in Configure System.
- Duplicate the device carrier and assign it to the correct driver which was duplicated.
- Set the correct slot ID of the driver in Configure System.
- Place the carrier onto the worktable.

- Execute the Device Placement.



For non-grid based devices like e.g. readers and washers that are usually screwed directly onto the instrument and are not placed onto another reference position, the above mentioned process doesn't work, since there is no nominal position available. Instead these devices are positioned or taught directly into the global coordinate system. This is done by using the "Placement" section in the Carrier editor (["see chapter 8.2.3" \[118\]](#)).

Also see about this

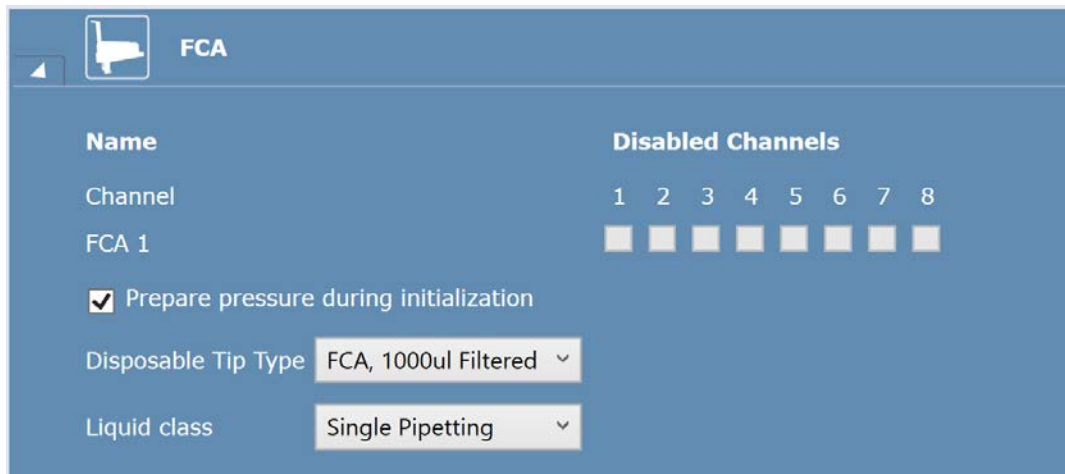
- ["Automatic Site Adjustment" ▶ 125](#)
- ["Positions / Teaching" ▶ 130](#)
- ["Robot Vectors" ▶ 121](#)
- ["Manually Mount Tool \(FCA\)" ▶ 175](#)

9.2 Configuring Your Drivers

In the Available Drives section you find a list of all available drivers. By clicking the relevant driver checkbox the driver software is enabled. This is only applicable if the assigned device is installed and placed on the deck.

Select the I/O State check box for all devices which are installed and physically connected to the PC vControl is running on. If the I/O is on, vControl will try to interact with the device or module at run time. If the I/O is turned off, the commands are still visible in the software (for most but not all drivers) and vControl will not interact with the device or module at run time.

9.2.1 FCA Configurations



The screenshot shows the 'FCA' configuration window. It has a title bar with a gear icon and the text 'FCA'. Below the title bar, there is a 'Name' field with the value 'Channel'. To the right of the 'Name' field is a 'Disabled Channels' section with eight checkboxes labeled 1 through 8. Below this, there is a checkbox labeled 'Prepare pressure during initialization' which is checked. Below that is a 'Disposable Tip Type' dropdown menu with the selected value 'FCA, 1000ul Filtered'. At the bottom is a 'Liquid class' dropdown menu with the selected value 'Single Pipetting'.

This section contains configuration settings for the FCA.

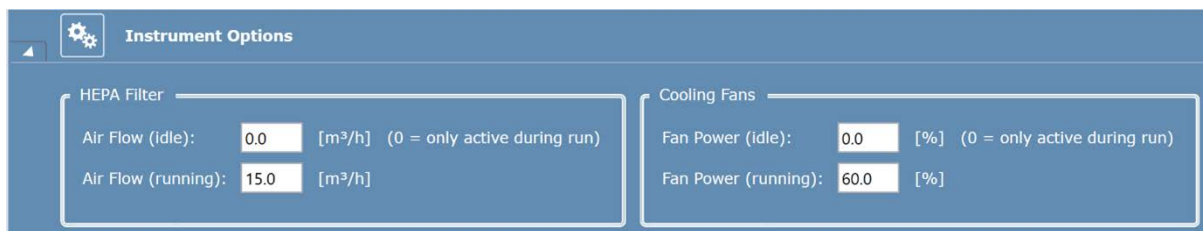
Disabled Channels: If a channel has been deactivated (marked as “broken”) due to an error such as a failed channel self-test, it can be activated again by deselecting its corresponding checkbox.

Prepare pressure during initialization: This option allows preparing the arm for a subsequent pipetting workflow as part of the initialization sequence. Since pressure preparation may consume a certain amount of time, using this option allows the system to build up the pressure while still being occupied with other tasks.

Disposable Tip Type: Needed for the option “Prepare pressure during initialization”. The tip type which is planned to be used for the subsequent pipetting workflow.

Liquid Class: Needed for the option “Prepare pressure during initialization”. The liquid class intended to be used for the subsequent pipetting workflow.

Instrument Options



The screenshot shows the 'Instrument Options' configuration window. It has a title bar with a gear icon and the text 'Instrument Options'. Below the title bar, there are two main sections: 'HEPA Filter' and 'Cooling Fans'. The 'HEPA Filter' section has two input fields: 'Air Flow (idle):' with a value of '0.0' and '[m³/h] (0 = only active during run)', and 'Air Flow (running):' with a value of '15.0' and '[m³/h]'. The 'Cooling Fans' section has two input fields: 'Fan Power (idle):' with a value of '0.0' and '[%] (0 = only active during run)', and 'Fan Power (running):' with a value of '60.0' and '[%]'.

This section contains configuration settings for different instrument options.

HEPA Filter: Configure the air flow of the HEPA filter fans.

Air Flow (idle): Configure the air flow of the fans when the instrument is idle (i.e. during Edit Time). A value of 0 indicates that the fans shall only be active during a run.

Air Flow (running): Configure the air flow of the fans during the run. The system will monitor the fans and raise a warning if their performance is out of an expected range.

Cooling Fans: Configure the power of the cabinet cooling fans.

Fan Power (idle): Configure the power of the fans when the instrument is idle (i.e. during Edit Time). A value of 0 indicates that the fans shall only be active during a run.

Fan Power (running): Configure the power of the fans during the run. The system will monitor the fans and raise a warning if their performance is out of an expected range.

9.2.2 Inheco Device Configurations

MTC Controlled Devices

The following inheco devices are controlled by STC/MTC and the setting schema is valid for all:

- CPAC
- CPAC2TEC
- Teleshake 95 AC
- Thermoshake AC

Slot number: represents the slot, on which port is the device connected to the MTC/STC controller.

Abort the incubation automatically, if the run is aborted: if the check box is activated the current incubation on the device will be aborted in case of run abort, otherwise the incubation continues.

Plate Incubator

Temperature Settings

Initial Temperature: after a successful initialization the incubator will set to the given temperature. Range: from ambient temperature to 80°C.

Tolerance Range: this range will be used in the “Wait For Temperature” command to check if the given temperature has been reached.

Shaker Settings

These values will be used if a robotic arm transports a labware into the incubator, the device will start shaking with the given parameters until the labware will be moved out from the device. If the shaking speed = 0, the device will not start to shake until an explicit Start Shaking command will be used.

Shaker Speed: defines the shaking speed in Hz, allowed range 0 and from 6.6 to 30.0 Hz.

Shake Strategy: this value determines the shaking direction, like: East-West, North-South or Round .

Slot Configuration

Number of Slots: determines how many incubators/slots shall be controlled by this driver.

The value >1 is relevant if multiple inheco incubators connected to each other via daisy chain (Master/Slave or Slave/Slave).

Location Group Name: name of the site from the carrier, where the incubator is defined as device.

Position: number of the site (read only)

Slot Address: this number will be used to define the device index and it is relevant if the driver shall control more than one devices.

9.3 Other System

The Configure System menu provides general settings as well as driver-specific settings.

- **Dialogs:** Shows all suppressed dialogs and allows to un-suppress them again.
- **Sound:** A run can finish either successfully, with warnings or errors. The acoustic sound for each of the options can be set. Pause:
- **Colors:** Default color coding for different drivers and command groups is already activated. This can be changed here and for non-default drivers this can be set.
- **Timeouts:** Specifies the timeout time for different actions. Initialization, Shutdown, Standby and Instrument Ready. If a 0 is specified then there is no timeout.
- **Instrument Configuration:** Select here the instrument configuration when in simulation mode. When switching to simulation mode without ever having been connected to a real instrument the software takes the first one selected in the dropdown list. When a different configuration is needed, this can be changed here. After changing the instrument configuration a software restart is required. This will be prompted by the software after saving.
- **Application Settings:** In this sections are some collected options related to run behaviour of the software.
- **Diagnostics:** This settings allow to adjust diagnostic file collections.
- **DeckCheck:** DeckCheck -specific settings. Refer to chapter DeckCheck for details.
- **LogLevel:** Provides options to collect more or less specific diagnostic information on the system.
- **Scheduler:** Refer to chapter Scheduler for more details.

- **Home Positions:** Here for each arm the parking positions for the idle mode can be defined. To set the positions either move the arm manually or automatically to the desired position, then press Transfer button. The values will update accordingly. The arm moves to this position either before shutdown of the software and/or at the end of every run. FCA-specific settings.

10 Validation Checklist

To ensure against malfunction and incorrect results, all vControl runs must be validated before they are used for pipetting with real samples. To ensure that all your runs operate as intended, it is highly recommended to perform a test run. This topic gives an overview of some important points to be checked.



The following validation checklist is not intended to be exhaustive and does not replace any local laws or statutes that apply.

Important points to be considered when validating vControl applications:

Tab. 3: Checklist for validating vControl applications

Done (x)	Task
	Check the Log File: Start vControl and, before you start pipetting, check the log file for errors. If there are error messages, the cause of all errors must be remedied first!
	Instrument: Make sure that the status bar lists the Instrument Mode as Real. STEP RESULT: In simulation mode, the system will not pipette any liquid! Do not use the simulation mode to validate a run. Make sure that the status bar lists the Instrument Mode as Real.
	Make sure that the correct arms and devices are used during the run.
	Make sure that the script or method is saved and reloaded before you start the validation.
	Make sure that you have loaded the correct script or method for the assigned pipetting task.
	Verify that Context Check is enabled in the Run menu and check the script editor window for errors (commands which have errors are shown in red).
	Check the liquid containers and system liquid of the pipetting instrument for the correct liquid types and concentrations which are required by the script or method.
	Before running the script or method, you must physically set up the worktable with the carriers, labware and devices which are needed by the application or assay. The administrator or owner of the pipetting instrument must establish standard operating procedures for setting up the worktable and make them available to everyone who carries out pipetting.
	After setting up the worktable with the carriers, labware and devices which are needed by the application or assay, inspect the paths that the instrument arms (FCA and RGA) will take during the script/method run to make sure that there are no obstacles which could obstruct the movements.

Done (x)	Task
	<p>Observe the arm initialization at the pipetting instrument:</p> <ul style="list-style-type: none"> On the Run menu, click Initialize Instrument. Make a visual check that the initialization sequence has been carried out correctly. Check the log file for initialization errors. ADDITIONAL INFORMATION: If error messages are found, the cause of the error must be remedied before you start pipetting.
	<p>Before you pipette using real samples, test run the script or method to make sure that:</p> <ul style="list-style-type: none"> The correct liquid classes are used for pipetting. Pipetting takes place at the intended positions on the worktable (e.g., correct micro plate, correct well). The correct volumes of liquid are pipetted. Pipetting takes place in the correct sequence as required by the application. Plate movements are as expected and with no visible drops or collisions. Plate movements from storage locations (e.g., stacker, carousel, hotels) fetch the expected plates and deliver them to the expected positions. Taking objects to waste positions functions as expected (with either the FCA or RGA) and that the waste chute/position remains open and accessible throughout the run.
	<p>If your script employs Worklisting, validate the creation of the input files and that the expected pipetting positions are used at run-time with differing input files.</p>
	<p>If you are using a Variable Pipetting Loop or variables to impact the number of samples in a run, verify that the correct number of samples is pipetted.</p>
	<p>If your script uses cLLD functionality (liquid level detection, aspiration supervision, retract supervision), make sure that the related error handling is executed correctly (as selected in the liquid class or the LLD Error Handling script command).</p> <ul style="list-style-type: none"> To ensure accurate liquid detection, make sure that the labware is correctly seated on the carriers. To verify that the cLLD system is operating, execute the cLLD Self Test direct command. NOTE: The capacitive detection system is not suitable for all liquid types (e.g., liquids with foam, non-conductive liquids).

Done (x)	Task
	When setting up an assay for a liquid which can clot, make sure that retraction supervision is enabled in the liquid classes which are used to handle the liquid. Check the execution of the retraction to verify that the tips are outside the liquid afterwards. Refer to the Good Usage information in the Operating Manual for the pipetting instrument for additional recommendations.
	If your script uses septum piercing, make sure that the piercing movement is performed correctly.
	If your script uses arm movement commands, make sure that the arm moves to the correct position.
	Do avoid firmware commands. If your script or method sends firmware commands to the pipetting instrument or to other devices, make sure that the correct firmware commands have been specified and that they initiate the intended actions. Also verify that the results are as expected as firmware commands are not tracked by modules such as Sample Tracking.
	If your script or method uses a microplate reader, check the results (output data) from the reader to make sure that they are plausible.
	Make sure that all liquid transfers work correctly. When setting up an assay which uses a trough (container for a reagent liquid), make sure that the liquid classes which are used for aspirating from the trough are configured to immerse the tips in the liquid by at least 1 mm.
	Verify that the washing actions are executed correctly. Appropriate tip washing actions must be provided in the script or method to avoid cross-contamination of samples and/or reagents. Disposable tips can also be used to avoid cross-contamination of samples and/or reagents. Design and test run the script or method to make sure that cross-contamination will not take place.
	When using disposable tips, make sure that: <ul style="list-style-type: none"> • The tips are correctly picked up and dropped / set back. • The correct labware is used for picking up and dropping / setting back tips.
	If your script uses a barcode reader, make sure that the barcodes are scanned correctly and are assigned to the correct labware. This includes standalone readers or those integrated in devices like a Stacker or Carousel.
	If your script uses a robot to transport labware, make sure that the robot paths are executed correctly.
	If your script or method uses a shaker or an incubator that shakes, make sure that the shaker parameters (e.g., shaking time, shaking frequency) have been configured correctly and that the shaking functions as intended.

Done (x)	Task
	If your script or method uses an incubator, make sure that the incubator parameters (e.g., incubating time and temperature) have been configured correctly and that the incubator operates as intended, including if applicable the functionality of the doors.
	If your script or method uses devices with heating elements (e.g., shaker, heated incubator, some microplate readers), make sure that the script or method allows enough time for the heater to heat up to or cool down to the intended temperature before it is used. Test run the script or method and measure the temperature which has been achieved.
	If your script or method uses operations where timing is critical (e.g., incubator, shaker), make sure that the timing and delays have been configured correctly and work as intended: Check the time specified in the respective device command. Check the time specified in Wait For Timer commands.
	If the system pauses for any reason when running a script or method, the duration of the pipetting script may differ from the intended time. For example, sample incubation time may be outside of the specified limits. Depending on the application, this can make the assay invalid. A system pause can be caused for example by clicking the Pause button in vControl's Runtime Controller dialog box, by pressing the Pause button on the pipetting instrument, by a pipetting error or by an internal timeout if a step has taken longer than expected or specified.
	If your script uses the Touch Tools RUP script command, make sure that: <ul style="list-style-type: none"> The contents of the prompt are displayed correctly. Modifications to variables are executed correctly.
	If your script or method uses e-mail for notification of important events, the notification function must be used to send a test e-mail to make sure that the e-mail is sent correctly and is received by the intended recipient.
	If your script or method imports required information from external files (bar code lists, lists of values for variables, etc.): Make sure that the correct file name has been specified in the script or method. Make sure that the import file exists and contains the intended data.
	If your script exports information (e.g., variable values) to files, make sure that the export file has been generated correctly and contains the correct data.
	If your script uses variables or expressions, make sure that the correct values are used during the run.
	Check that the script or method starts to run as intended when you click the Start button in the Runtime Controller or touch interface.
	If your method consists of more than one batch or contains several scripts, check that the batches and scripts are executed in the correct order.

Done (x)	Task
	<p>If your script uses subroutines, check that:</p> <ul style="list-style-type: none">• The subroutines are loaded and executed correctly.• Concurrent arm movements are performed correctly and without collisions.
	<p>When the script or method has completed, check the status information and the Warnings / Errors window in Runtime Controller and Trace View, and check all log files to make sure that the entire script or method has completed and that there are no error messages.</p>
	<p>The results of an assay are only valid if the entire script or method completed without errors. If this is not the case, the entire script or method must be repeated after rectifying the cause of all errors.</p>
	<p>If your script or method uses Sample Tracking to generate a report, verify that all values and actions performed (e.g., pipetting and transferring labware) are correctly reported and assigned to the expected labware and wells.</p>

11 Commands

This chapter contains detailed information about the usage of commands and smart commands. Smart commands are used to program liquid handling workflows for the FCA and MCA arms and comprises the following commands: **Sample Transfer (FCA)**, **Reagent Distribution (FCA)**, **Standard Curve (FCA)**, **Transfer Individual Volumes (FCA)** and **Transfer Liquid (MCA96 and MCA384)**. They are able to create so called smart routines, while normal commands cannot do this.

11.1 Direct Commands

Direct commands are used to execute an action without running a script line (e.g., for picking up DiTis or flushing the tips). They provide a convenient alternative to creating and running a pipetting or maintenance script that carries out the same function.

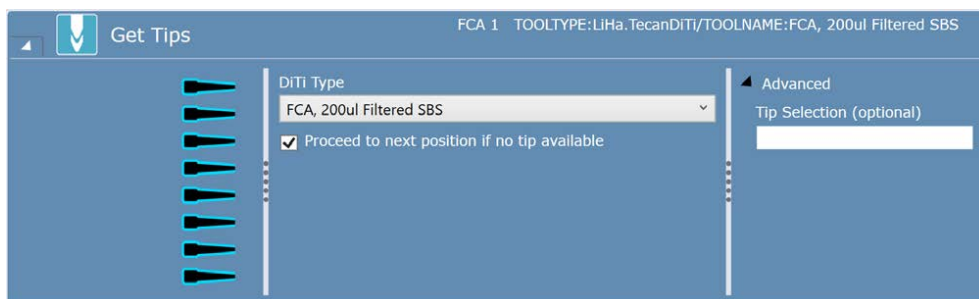
The Controlbar only shows direct commands for devices whose device driver is installed and configured(e.g., the DiTi commands are only shown if the FCA is configured with DiTis).

To run a Direct command, right-click on the command and then click Run direct in the shortcut menu. Alternatively, you drag the Direct command to the run area of the Start tab.



Direct Commands will use the worktable which is open in the Worktable Editor when the Direct Command is executed. If no worktable is open, the Direct Command will use the worktable that has been most recently used in a run, including the most recent DiTi positions. Before executing a Direct Command, make sure that the arm is outside a contouring-error prone position (i.e., bounding boxes).

11.1.1 Get Tips (FCA)



This command picks up disposable tips (DiTis) of the specified type from a DiTi rack. The application keeps track of the available tip positions on the worktable and automatically picks up the next available unused DiTis of the chosen type.

This information is also persisted when shutting down the application or powercycling the instrument/PC, and is used when executing a run with “stored tip positions”.

Order of Tip Pickup

If there is more than one tip box containing tips of the selected type, pickup starts at the box with the lowest index number (e.g., FCA_200ul[001] before FCA_200ul[002]). In general tips are picked up from left to right within a box, but preference is given to fast pickup over strict left-to-right use—e.g.: 8 tips will always be picked up from a full column and not in two steps starting from an incomplete column in a tip box.

DiTi Type

Click the arrow in the dropdown list to see more options and select the DiTi type of the FCA arm from the list. The tip type can also be selected by clicking a tip box on the worktable.

Proceed to next position if no tip available

If no DiTi is present at the current position and this option is selected, the software automatically moves to the next position. If a tip is detected but cannot be successfully picked up an error message is shown. Follow the instructions of the error message.

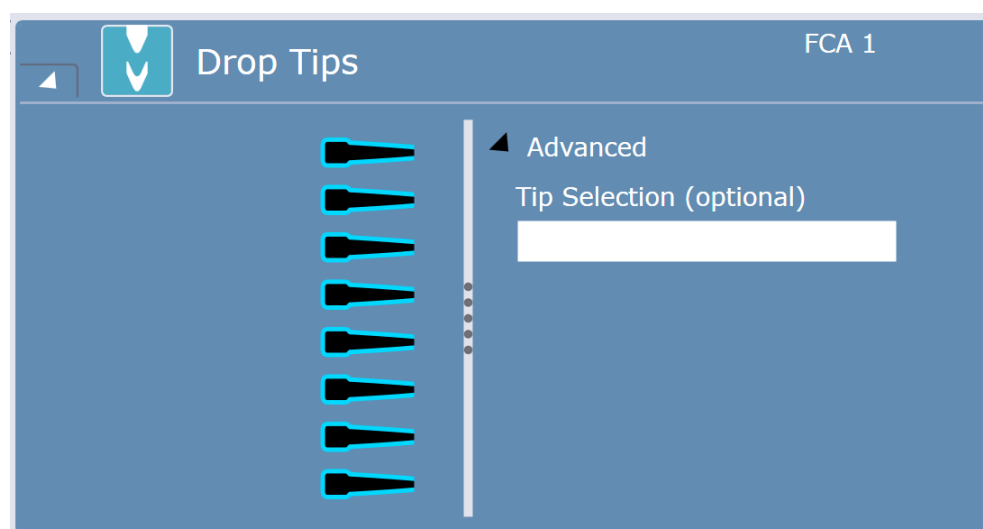
Advanced Parameters

Tip Selection (optional):

The Tip Selection (optional) parameter is resolved as integer value (integer variables and expressions are allowed) and then interpreted as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

11.1.2 Drop Tips (FCA)



This command discards disposable tips (DiTis) from the selected channels into a DiTi Waste.

The software automatically chooses a suitable waste labware on the worktable.

11.1.3 Pick Up Tips (FCA)

Robotic Actions: Prior to execution of the script command, the system performs the following:

- If parameter **Airgap Volume** is set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates an air gap of volume 20 µl.
- If parameter **Airgap Volume** is not set to zero and the last pipetted volume in variable **tipContentVolumes** is zero, the system aspirates the air gap as defined by the **Airgap Volume** parameter.

Parameters

Tip Selection: Select the tips you want to use. If you do not click a tip, vControl uses all tips by default. To turn off or turn on a tip, click on it.



To avoid errors at run time, the Pick Up Tips command should not be used to fetch tips from nested or stacked labware. The Get Tips command must be used instead.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Advanced Parameters

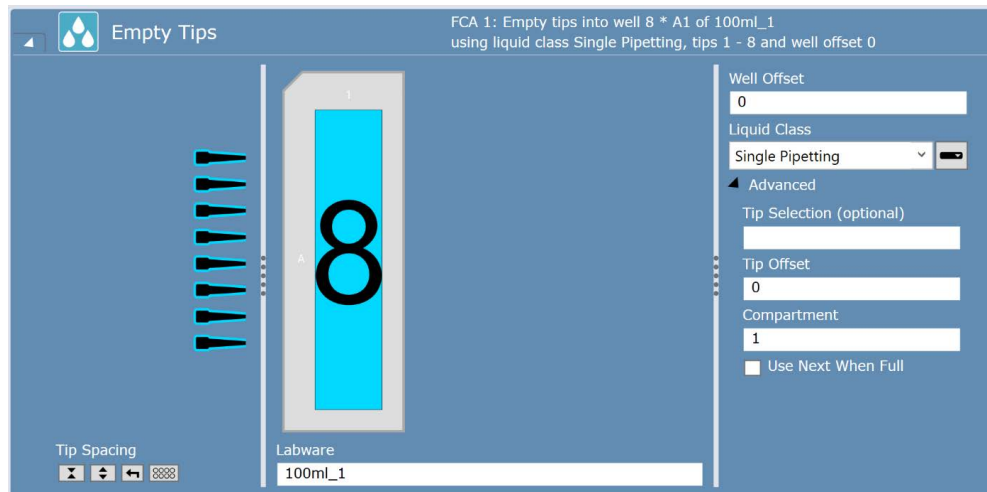
Tip Selection (optional): For this parameter definition, follow the cross reference to Tip Selection (optional).

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Air gap volume [µl]: The volume for the air gap in microliters.

Airgap Speed [µl/s]: In the Airgap Speed box, enter the air gap speed you want to use in microliters per second.

11.1.4 Empty Tip (FCA)



This command dispenses the entire content of the selected tips into specified positions on the worktable.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.

- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support the empty tips workflow.

Advanced Parameters

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

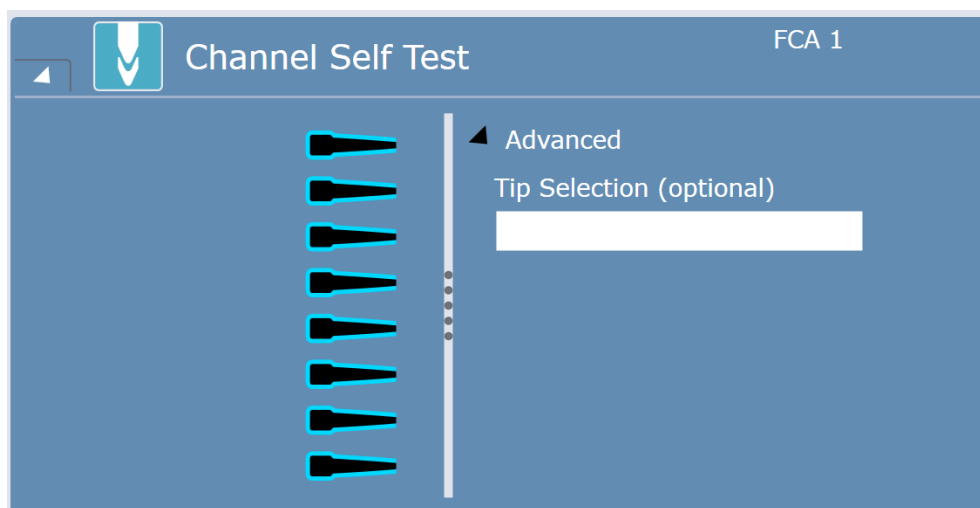
X offset [mm]: This is the offset of the tip from the center of the target well (negative = to left, positive = to right).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Use next when full: If this option is selected, the software performs the following checks before executing the dispense:

If the selected labware contains the custom attribute **LiquidWasteCapacity**, the software checks whether the remaining capacity (i.e. the **LiquidWasteCapacity** minus the current volume in the cavity) is sufficient for the volume to be dispensed. If the remaining capacity is insufficient, the software will automatically switch to the next labware in the group (e.g. if "Trough[001]" is full, it will choose "Trough[002]"). If all labware items in the group are full, the software will prompt the user to empty them and start with the first selected labware again.

11.1.5 Channel Self-Test



This command performs a test of the inline filter of the FCA channels to detect potentially damaged or wet filters.

In order to protect the pipetting channels against over-aspiration of liquid (e.g. if a disposable tip with smaller capacity than intended is used) a control system is installed on each channel. The control system combines a filter inside the DiTi cone and the pressure monitoring of the channel.

This command can only be executed without mounted disposable tips.

Parameters

Tip Selection: Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

The software automatically executes a channel self-test in case of a pressure profile error during pipetting. In that case, the self-test is performed as part of the Drop Tips command, after the tips have been discarded from the affected channels

If the self-test fails, the affected channel is deactivated for further usage (marked as "broken" in the configuration).

Channels can be activated again by passing an inline-filter test. If an inline filter is damaged a service technician is needed to replace the inline filter.

11.1.6 Get Fingers (FCA)

Use this command to pick up the FCA gripper fingers. Refer to section Get Fingers.

11.1.7 Drop Fingers (FCA)

Use this command to place the FCA gripper fingers to a specified position on the deck. Refer to section Drop Fingers.

11.1.8 Manually Mount Tool (FCA)

This command can be used to mount a reference tip manually on to a FCA channel. The user will be prompted to perform the mounting. After confirmation the state will be reflected in the application.

11.1.9 Manually Unmount Tool (FCA)

This command can be used to unmount a reference tip manually from a FCA channel. The user will be prompted to perform the unmounting. After confirmation the state will be reflected in the application.

11.1.10 Fetch Labware for Teaching (RGA)

Use this command to pick up labware and carriers in order to teach sites with the RGA.

Carriers When teaching sites with the RGA, the carrier name has to be typed in manually. The carrier needs CustomAttribute **MoveableCarrier** set to true and to specify CustomAttributes **Grip_Narrow** and **Grip_Wide**. **24 Eppendorf Adapter** is a moveable carrier and can be used as example.

Parameters **Labware:** In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

11.1.11 Set Available Tips (Worktable)

Use the **Set Available Tips** command to set the next available position to fetch disposable tips from.

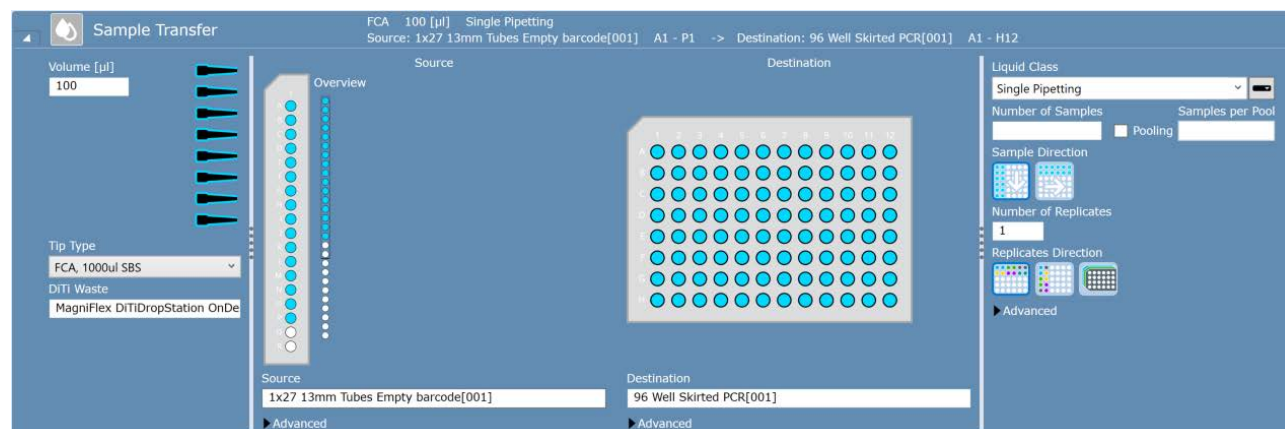
1. First select the labware containing the DiTis (see Labware parameter below).
2. Use the following wildcard characters while typing in the Labware text box:
** denoting an arbitrary sequence of characters*
? denoting one arbitrary character
3. Select the next available DiTi position in the labware.
4. Mark the positions of the next available DiTi in the plan view of the labware layout.

vControl will update the next available DiTi position for all labware items matching the labware name defined by the user.

Parameters **Labware:** In the **Labware** box, either type your expression or click the labware displayed in the Worktable editor window (*Worktable click*).

11.2 FCA

11.2.1 Sample Transfer (FCA)



The liquid handling arm provides script commands to distribute reagents and transfer samples including the picking up and dropping of disposable tips.

Instead of a series of Aspirate and Dispense commands, you can use this smart command to program a sample transfer based on the number of samples from a selected source labware to a destination labware.

A common workflow in the lab is to transfer samples from many tubes to one or more plates. With the Sample Transfer smart command, you have an easy and performant way to program a sample transfer based on the number of samples, which may be contained in multiple source labware, to destination labware. In the destination labware you are able to define where the samples may be pipetted, within this range the samples may be pipetted row-wise or column-wise. Optionally, you can define the number of replicates and if they should be pipetted row-wise, column-wise or plate-wise

Parameters

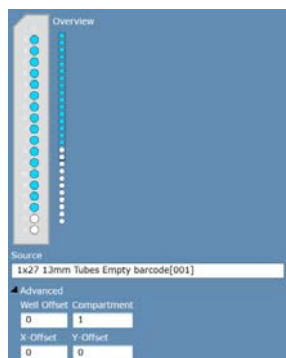
The Sample Transfer script command provides the following parameters for the configuration of the pipetting automation:

Volume [µl]: Specifies the volume to be transferred to each target well. If this volume is larger than the available tip capacity or the syringe volume, the system will transfer the volume in multiple equal steps at runtime.

Tip selection: Selects which pipetting channels of the modules are used, optionally a tip mask might be applied.

DiTi Type: Select the type of disposable tips to be used for pipetting from the dropdown list.

DiTi Waste: If the tips need to be changed (e.g., between samples), the DiTi Waste is the position which will be used to discard the tips.



Aspiration position: Selection from where aspiration is performed by definition of a source labware on the worktable, well selection and optionally (Advanced section below the labware) well offset, a general X/Y offset within the well and the compartment number.

Dispense position: Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection and optionally (Advanced section), well offset, a general X/Y offset within the well and the compartment number.

Liquid Class: Select the liquid class to be used for aspirate and dispense.

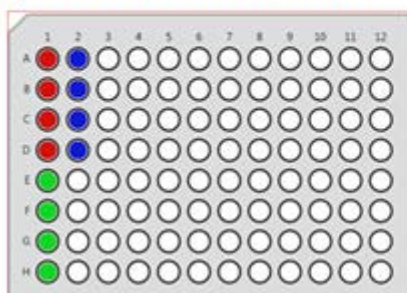
Number of Samples: Number of target wells to be pipetted. This parameter is optional; if left empty, the number of samples corresponds to the number of selected source wells.

Sample Direction: Defines if the sample should be dispensed with each subsequent sample in the next available well in the row or column. Specify whether you want column-wise or row-wise direction.

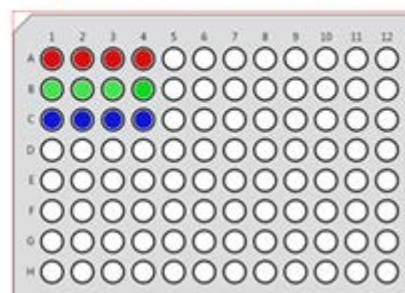
Number of Replicates: Allows replication of the selected samples. Default is 1.

Replicates Direction: Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise):

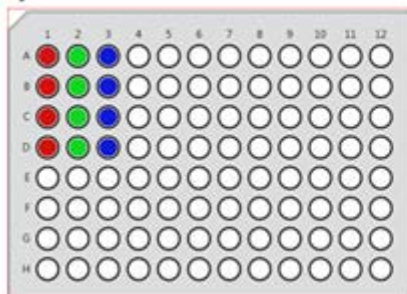
**Sample Direction: column,
Replicate Direction: column**



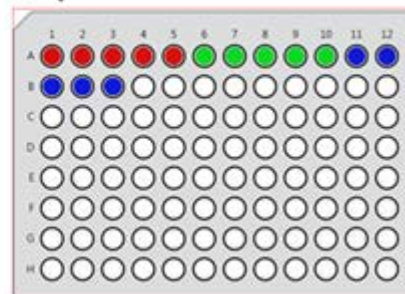
**Sample Direction: column,
Replicate Direction: row**



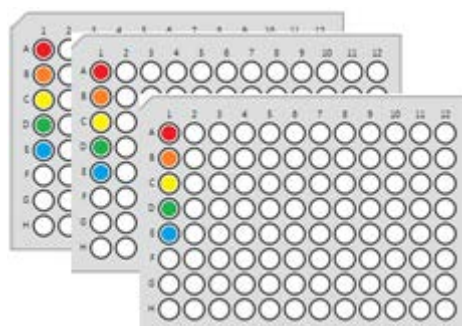
**Sample Direction: row,
Replicate Direction: column**



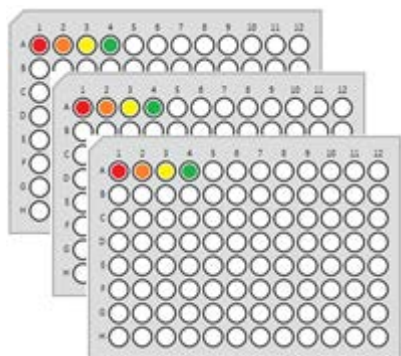
**Sample Direction: row,
Replicate Direction: row**



Sample direction and replicates direction (row-wise and column-wise and their possible directions)



Sample Direction: column-wise, Replicate Direction: plate-wise (Number of Replicates: 3)



Sample Direction: row-wise, Replicate Direction: plate-wise (Number of Replicates: 3)

If Replicates Direction row or column is selected, a replicate group (all replicates of a single sample) is kept within a contiguous range and pipetted into the same plate.

Advanced Parameters

Max Multi Dispenses: Maximum number of multi-dispenses.

Max Tip Reuse: Maximum number of allowed reuses (i.e. number of aspirations) allowed to be performed with an individual disposable tip.

Tip Selection: Optional tip mask to be applied to the tip selection in the UI.

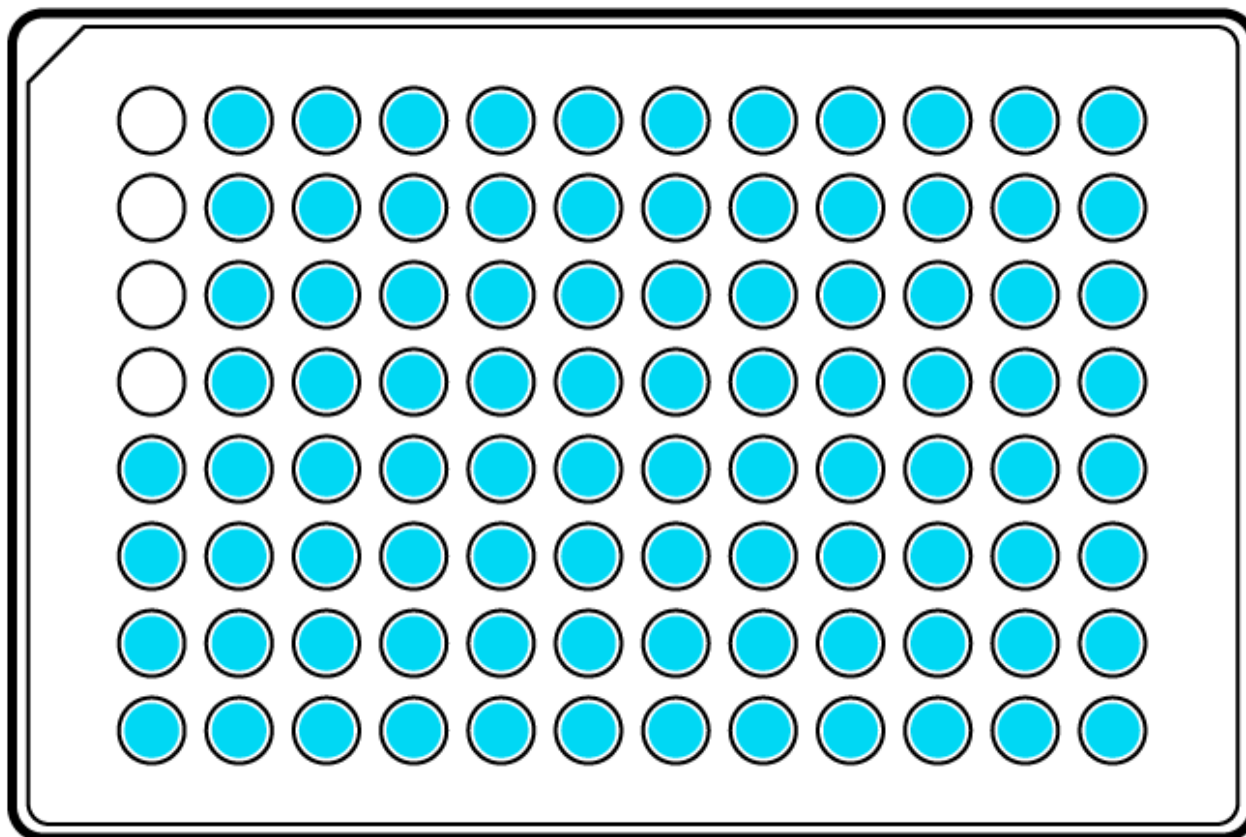
Tip Offset: Optionally shifts the tip selection in the UI.

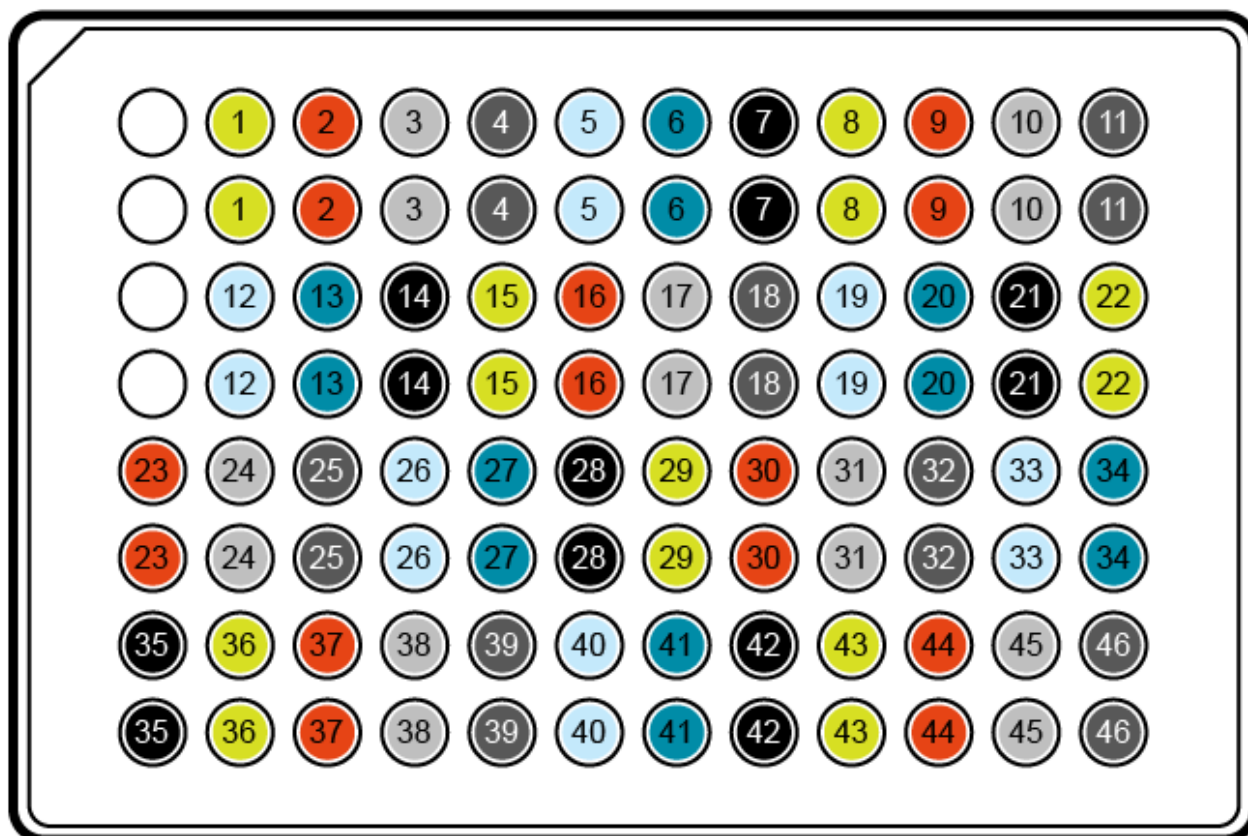
Airgap volume / Airgap speed: Volume and aspiration speed of blowout airgap, aspirated when picking up disposable tips.

Empty Tip Content Liquid Class: Liquid class to use when performing an Empty Tips (used in multi pipetting sequences to discard the excess volume).

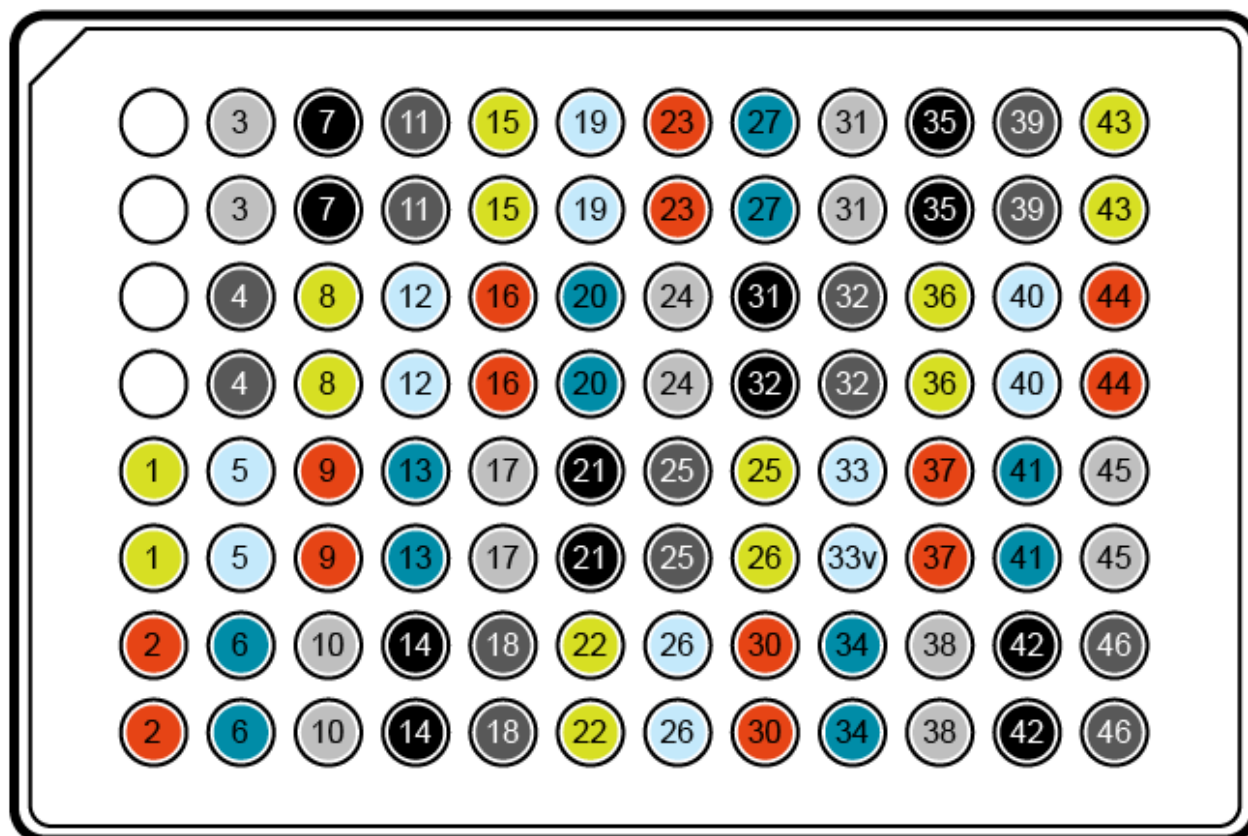
Empty Tip Content Labware: Labware to use when performing an Empty Tips, usually a liquid waste.

Example plate
map (gaps for
controls)

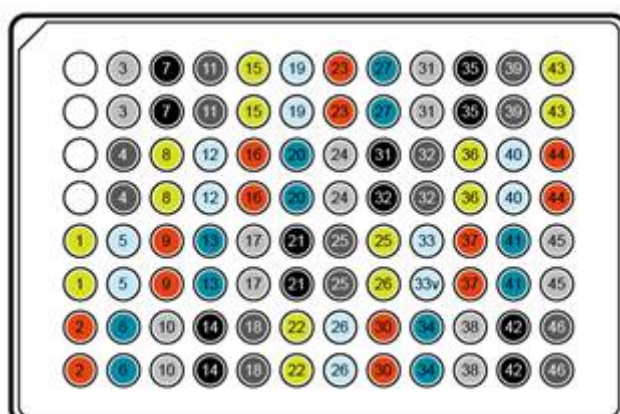




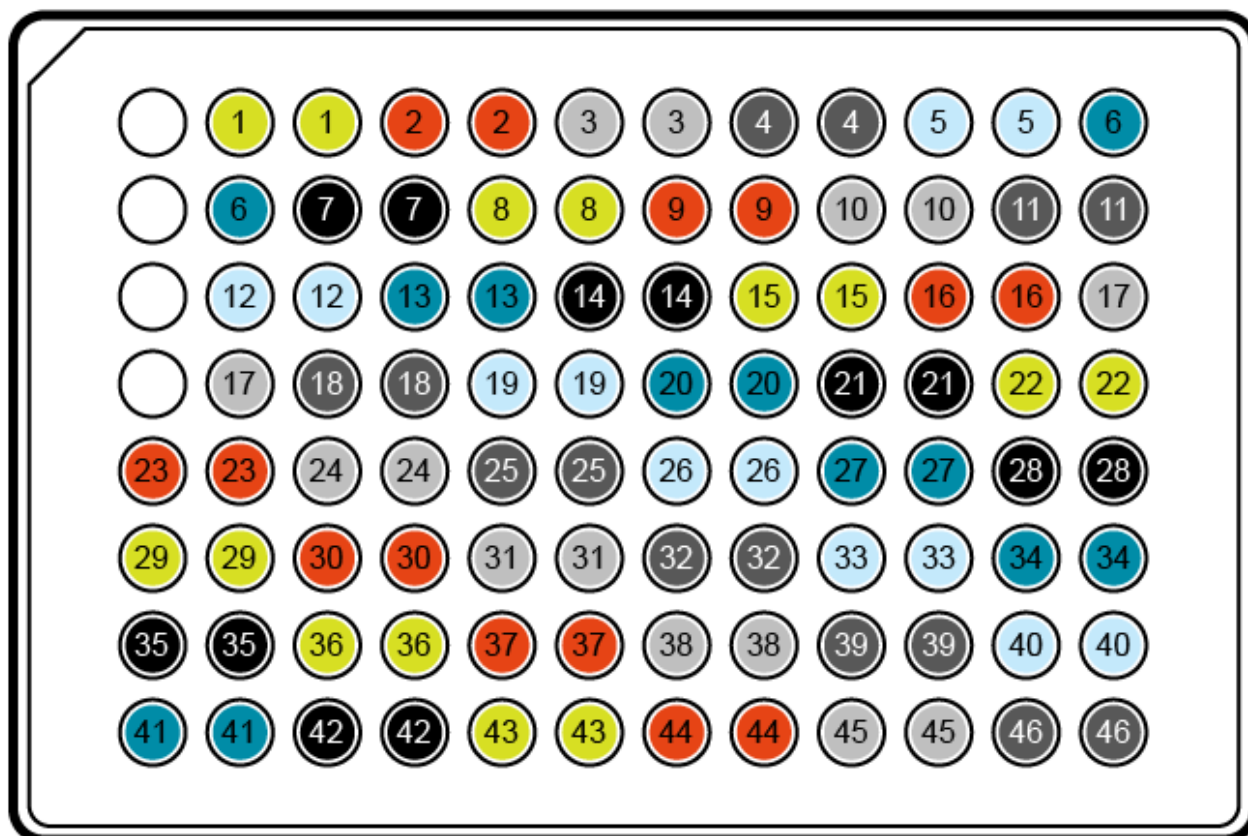
Selection range leaving 4 wells empty for controls



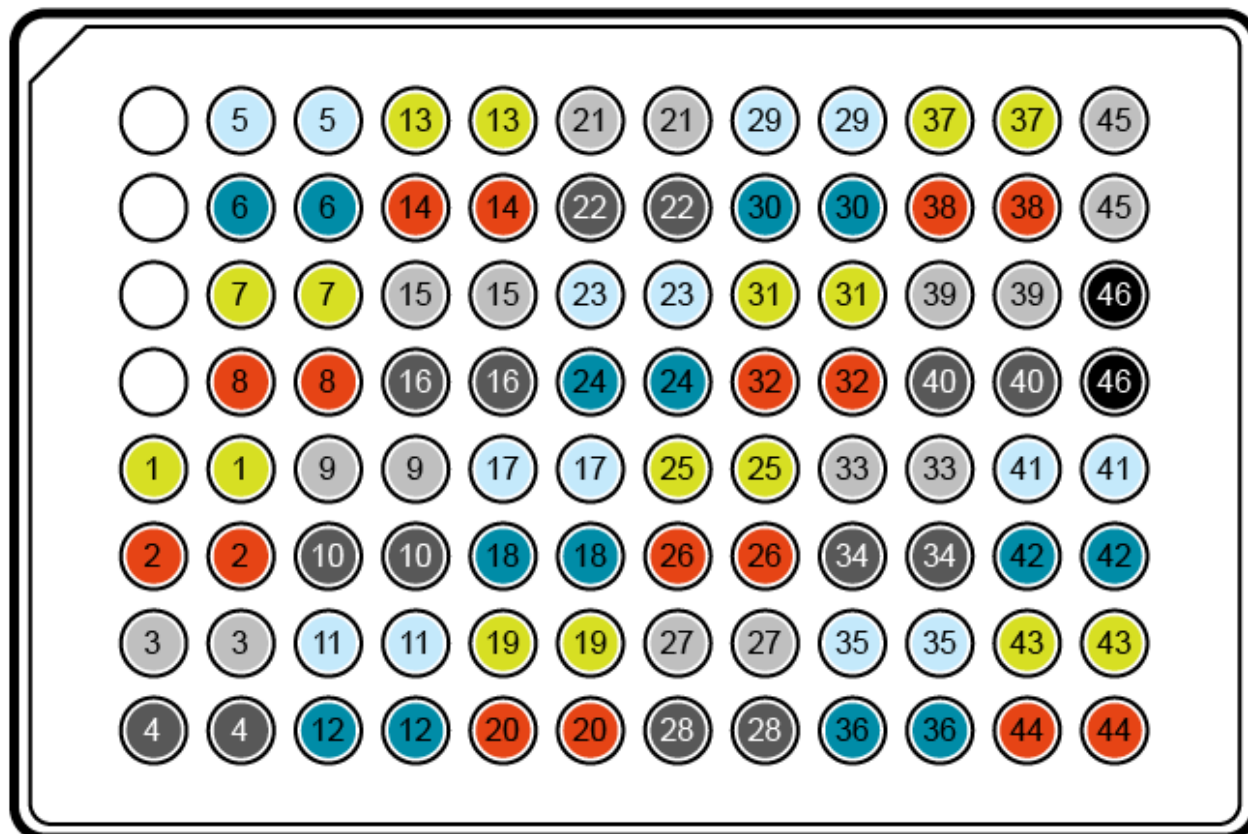
Samples row-wise, 2 Replicates column-wise and 4 empty wells for controls



Samples column-wise, 2 Replicates column-wise and 4 empty wells for controls

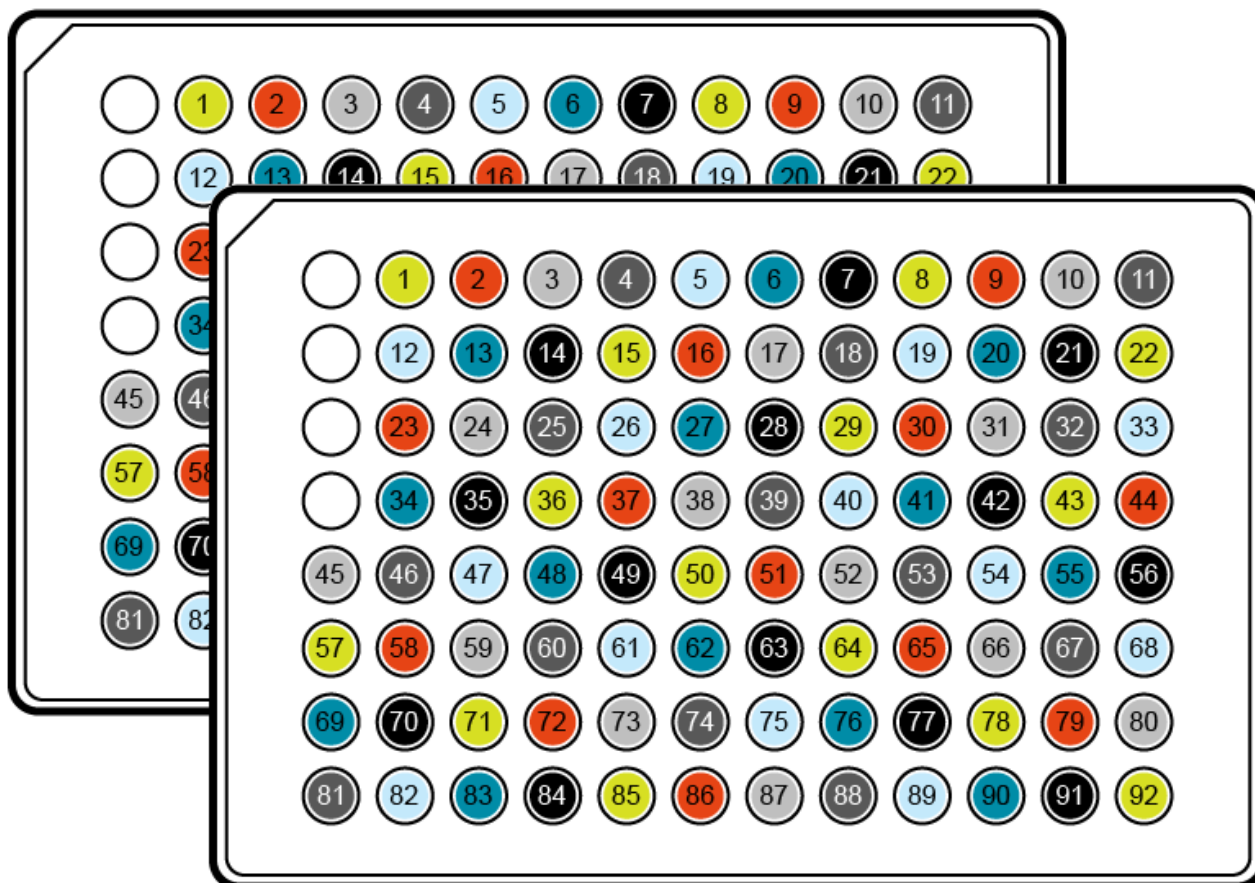


Samples row-wise, 2 Replicates row-wise and 4 empty wells for controls



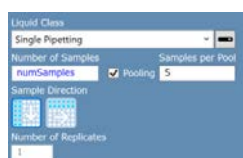
Samples row-wise, 2 Replicates row-wise and 4 empty wells for controls

Samples column-wise, 2 Replicates row-wise and 4 empty wells for controls



Samples row-wise, 2 Replicates plate-wise and 4 empty wells for controls

In addition to a standard Sample Transfer workflow with (optional) replicates, the command also supports a Pooling workflow, i.e. the combination of a number of samples in one pool.



When checking the Pooling option, the number of Samples per Pool has to be entered. The software will then pipette contiguous groups of samples to one pool, i.e. the first group of n samples goes to the first pool (first selected target well).

If the total number of samples is not a multiple of the number of samples per pool, the software will issue a context check warning that the last pool contains fewer samples.

Note: When selecting the Pooling option, replicates and tip reuse are not supported and the corresponding input fields are grayed out.

11.2.1.1 Standard Curve

You can quickly program a pipetting sequence that pipettes a standard curve from a single source across a plate based on a plate map (gaps for samples), number of points in the curve and dilution options.

Common workflow in the lab is that it is necessary to know the exact concentration of the liquids that are handled. Especially before normalization the concentration of the individual liquids must be known.

A standard curve is used in quantitative research as reference to identify unknown concentrations of substances (e.g., DNA, protein). Multiple samples with known properties are measured and graphed, which then allows the same properties to be determined for unknown samples by interpolation on the graph.

A key operator wants to be able to quickly program a pipetting sequence pipetting a standard curve from a liquid with known concentration.

With the Standard Curve script command, you can define the pipetting steps

Pipetting a standard curve from one source across parts of a target.

The script command allows configuration of the following for the pipetting automation:

1. Define the curve points and dilution ratios
2. Define one source well
3. Define diluent location
4. Define target wells
5. Define target layout
6. Define pipetting parameters as liquid classes and tip types

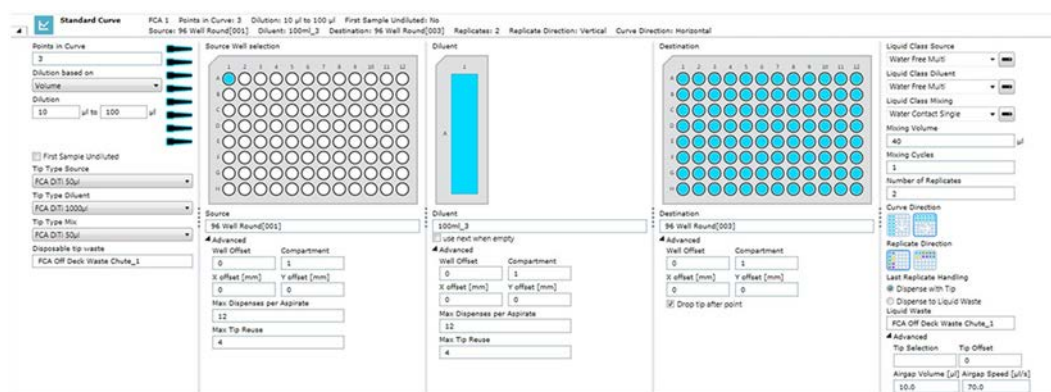


Fig. 87: Standard curve script command

The system allows selecting labware by clicking on the worktable.

1. Define the source labware (Source)
2. Define the diluent labware (Diluent)
3. Define the destination labware (Destination)
4. Click on a DiTi to define the TipType (DiTi Type)
5. Click on a Waste to define the Waste (DiTi Waste)
6. Click on a Cleaner to define the Cleaner

- There are different parameters available if you select Fixed Tips (the parameters for a wash cycle instead of a Disposable Tip Drop).
- Points in Curve

Specify the amount serial dilution steps of the curve.

- Dilution based on

Specify the ratios that count for each point inside the curve.

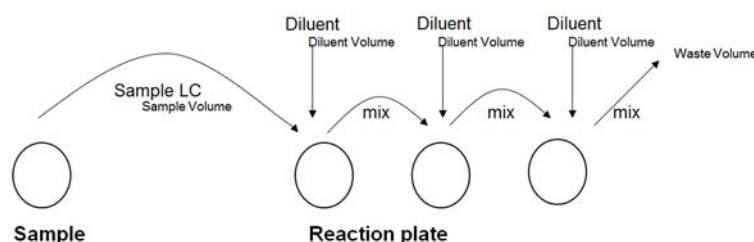


Fig. 88: Defined ratio

- Volume

Specify the exact volumes for the source and the diluent. The diluent volume will be the target volume.

- Factor

Specify the dilution factor—e.g.: 1:3 means 1 part sample and 2 parts diluent. Additionally, the target volume has to be specified. Target volume will be the volume of the dilution.

- Tip selection (click)

Selects which pipetting channels of the modules are used.

- Tip Type Selection (Source, Diluent, Mix)

Specify the tip types that shall be used for pipetting.

- Disposable Tip waste

If script command is used with FCA disposable tips:

Specify the Name of the Disposable Tip waste on the worktable which can be added via a worktable click.

- Liquid Waste

If script command is used with FCA standard tips:

Specify the Name of the liquid waste on the worktable which can be added via a worktable click.

- Cleaner

If script command is used with FCA standard tips:

Specify the Name of cleaner on the worktable which can be added via a worktable click.

- Liquid Class (standard tips)

If script command is used with FCA standard tips:

The liquid class that will be run for cleaning the standard.

- Volume in waste / in cleaner [ml]

If script command is used with FCA standard tips:

Volumes used to dispose liquid waste or clean FCA standard tips.

- Source well selection

Selection from where source aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well, compartment number, max. dispenses per aspirates and max tip reuse.

- Diluent well selection

Selection from where diluent aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well compartment number, max. dispenses per aspirates and max tip reuse.

- Use Next When Empty

If set the system uses the next available labware with the same name if the actual labware is empty.

- Destination

Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection, well offset, a general X/Y offset within the well and the compartment number. This option is not available for In Situ.

- Sample direction

Defines if samples are dispensed into the target labware in the next available well in the row-wise or column-wise. This option is not available for In-Situ.

- Replicate direction

Defines if replication direction is dispensed in the next available well in the row/ column or into the same positions of subsequent plates (plate-wise).

- Well Offset

Specify the offset inside a labware (source, diluent or destination). The wells until the offset will stay empty. In multiple plates this counts for the first plate. Counting is done top to bottom then left to right.

- Compartment

Defines the number of compartments of a well.

- X offset [mm]

X-Pipetting offset counted from the middle of the well. Allows pipetting next to the well border.

- Y offset [mm]

Y-Pipetting offset counted from the middle of the well. Allows pipetting next to the well border.

- Maximum dispenses per Aspirate

Number of dispenses allowed for pipetting until a reaspirate has to be done. After reaspiration or DiTi change this number starts counting from 0.

- Maximum Tip reuse

If script command is used with FCA disposable tips:

Maximum number of dispenses for an individual disposable tip

That is to say the maximum number of aspirations allowed to be performed with a disposable tip until it will be dropped.

- Tip Selection

Tip mask to define which pipetting channels are used.

- Tip Offset

Offset for tips stating which tip shall be used first. The tips until the offset will not be used for pipetting.

- Airgap Volume [μ l]

Volume of the airgap.

- Airgap Speed [μ l/s]

Plunger speed for the airgap.

11.2.2 Reagent Distribution (FCA)

With the Reagent Distribution script command, you can quickly program a pipetting sequence that transfers liquid (reagent) from a single source across one or multiple plate(s) based on a plate map (gaps for controls) and a number of samples.

Parameters

The Reagent Distribution script command provides the following parameters for the configuration of the pipetting automation:

Volume [μ l]: Specifies the volume to be transferred to each target well. If this volume is larger than the available tip capacity or the syringe volume, the system will transfer the volume in multiple equal steps at runtime.

Tip selection: Selects which pipetting channels of the modules are used, optionally a tip mask might be applied.

Tip Type: Select the type of disposable tips to be used for pipetting from the dropdown list.

Dynamic Tip Handling: Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

```
30:FCA DiTi 50µl;120:FCA DiTi 200µl
```

this means that: -DiTi 50 µl are used for volumes up to 30 µl, -DiTi 200 µl are used for volumes > 30 to 120 µl, and - for larger volumes the tip type selected in the Tip Type dropdown menu is used.

Tip Waste: If the tips need to be changed (e.g., between samples), the Tip Waste is the position which will be used to discard the tips.

Aspiration position: Selection from where aspiration is performed by definition of a source labware on the worktable, well selection and optionally (Advanced section below the labware) well offset, a general X/Y offset within the well and the compartment number.

Dispense position: Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection and optionally (Advanced section), well offset, a general X/Y offset within the well and the compartment number.

Liquid Class: Select the liquid class to be used for aspirate and dispense.

Number of Samples: Number of target wells to be pipetted. This parameter is optional; if left empty, the number of samples corresponds to the number of selected source wells.

Advanced Parameters

Sample Direction: Defines if the sample should be dispensed with each subsequent sample in the next available well in the row or column. Specify whether you want column-wise or row-wise direction.

Number of Replicates: Allows replication of the selected samples. Default is 1.

Replicates Direction: Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise).

Max Multi Dispenses: Maximum number of multi-dispenses.

Max Tip Reuse: Maximum number of allowed reuses (i.e. number of aspirations) allowed to be performed with an individual disposable tip.

Tip Selection: Optional tip mask to be applied to the tip selection in the UI.

Tip Offset: Optionally shifts the tip selection in the UI.

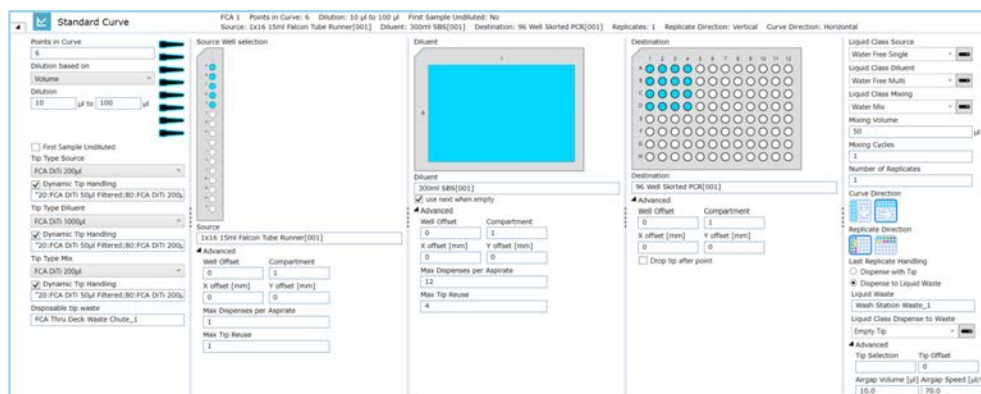
Airgap volume / Airgap speed: Volume and aspiration speed of blowout airgap, aspirated when picking up disposable tips.

Use Next When Empty: If this option is checked, the system will automatically proceed to the next source container in the same labware group if the first one has been emptied.

Empty Tip Content Liquid Class: Liquid class to use when performing an Empty Tips (used in multi pipetting sequences to discard the excess volume).

Empty Tip Content Labware: Labware to use when performing an Empty Tips, usually a liquid waste.

11.2.3 Standard Curve (FCA)



You can quickly program a pipetting sequence that pipettes a standard curve from a single source across a plate based on a plate map (gaps for samples), number of points in the curve and dilution options.

A common workflow in the lab is that it is necessary to know the exact concentration of the liquids that are handled. Especially before normalization the concentration of the individual liquids must be known.

A standard curve is used in quantitative research as reference to identify unknown concentrations of substances (e.g., DNA, protein). Multiple samples with known properties are measured and graphed, which then allows the same properties to be determined for unknown samples by interpolation on the graph.

The script command allows configuration of the following for the pipetting automation:

1. Define the curve points and dilution ratios
2. Define one source well
3. Define diluent location
4. Define target wells
5. Define target layout
6. Define pipetting parameters as liquid classes and tip types

The system allows selecting labware by clicking on the worktable:

- Define the source labware (Source)
- Define the diluent labware (Diluent)
- Define the destination labware (Destination)
- Click on a DiTi to define the TipType (DiTi Type)
- Click on a Waste to define the Waste (DiTi Waste)

Points in Curve: Specify the amount serial dilution steps of the curve.

Dilution based on: Specify the ratios that count for each point inside the curve.

- *Volume*
Specify the exact volumes for the source and the diluent. The diluent volume will be the target volume.

- **Factor**
Specify the dilution factor—e.g.: 1:3 means 1 part sample and 2 parts diluent. Additionally, the target volume has to be specified. Target volume will be the volume of the dilution.

Tip selection (click): Selects which pipetting channels of the modules are used.

Tip Type Selection (Source, Diluent, Mix): Specify the tip types that shall be used for pipetting.

Dynamic Tip Handling: Dynamic Tip Handling is designed to use different tip types depending on the aspirate volume. If you enter in the Dynamic Tip Handling box – e.g.:

```
30:FCA DiTi 50µl;120:FCA DiTi 200µl
```

this means that:

- DiTi 50 µl are used for volumes up to 30 µl,
- DiTi 200 µl are used for volumes > 30 to 120 µl, and
- for larger volumes the tip type selected in the Tip Type dropdown menu is used.

Disposable Tip waste: Specify the name of the Disposable Tip waste on the worktable which can be added via a worktable click.

Source well selection

Selection from where source aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well, compartment number, max. dispenses per aspirates and max tip reuse.

Diluent well selection

Selection from where diluent aspiration is performed by definition of a source labware on the worktable, well selection, well offset, a general X/Y offset within the well compartment number, max. dispenses per aspirates and max tip reuse.

Use Next When Empty: If selected, the system uses the next available labware within the same group if the actual labware is empty.

Destination well selection

Selection to where dispensation is performed by selection of the destination labware on the worktable, well selection, well offset, a general X/Y offset within the well and the compartment number. This option is not available for In Situ.

Sample direction: Defines if samples are dispensed into the target labware in the next available well in the row-wise or column-wise. This option is not available for In-Situ.

Replicate direction: Defines if replication direction is dispensed in the next available well in the row/column or into the same positions of subsequent plates (plate-wise).

Well Offset: Specify the offset inside a labware (source, diluent or destination). The wells until the offset will stay empty. In multiple plates this counts for the first plate. Counting is done from top to bottom, then from left to right.

Compartment: Defines the compartment within a well to be used.

X offset [mm]: X-Pipetting offset counted from the middle of the well. Allows pipetting next to the well border.

Y offset [mm]: Y-Pipetting offset counted from the middle of the well. Allows pipetting next to the well border.

Maximum dispenses per Aspirate: Number of dispenses allowed for pipetting until a reaspirate has to be done. After reaspiration or DiTi change this number starts counting from 0.

Maximum Tip reuse: Maximum number of aspirations allowed to be performed with an individual disposable tip.

Tip Selection: Tip mask to define which pipetting channels are used.

Tip Offset: Offset for tips stating which tip shall be used first. The tips until the offset will not be used for pipetting.

11.2.4 Transfer Individual Volumes (FCA)

The Transfer Individual Volumes command can be used to transfer different volumes from one labware to another. The Input of this command is a pipetting instruction file in the form of a CSV file, the separator is a semicolon (;). The user can create their own pipetting instruction file or use the generated pipetting instructions of the Normalization Calculation command. For more information on the Normalization Calculation command, refer to Normalization Calculation. The command will create a subroutine, which contains the optimal pipetting sequence to process the pipetting instruction input file.

Example

Source well index	Target well index	Volume [µl]
1	1	20.000
2	2	22.000

The raw text file of this example would then be:

```
Source well index;Target well index;Volume [ul]
1;1;20.000
```

```
2;2;22.000
```

The header must be included in the file and must contain these three columns.

Parameters

Tip Selection: Select the tips you want to use. By default the software uses all tips. To turn off or turn on a tip, click on it.

Reuse the mounted Tips and apply the sample filter: Selecting this checkbox will only process source wells from which currently mounted tips have aspirated.

Tip Type: This selection will define what type of disposable tip shall be used during the generated subroutine.

Dynamic Tip handling: The dynamic Tip handling is designed to use different Tip types depending on the aspirate volume, if these are available on the worktable. An example input of this field could be:

```
30:FCA DiTi 50µl; 120:FCA DiTi 200µl;
```

This would mean that:

- DiTi 50 µl are used for volumes up to 30 µl
- DiTi 200 µl are used for volumes > 30 µl to 120 µl

Tip Waste: This field defines where the used tips shall be disposed to.

Use the output from the normalization Command: Enabling this checkbox will use the pipetting instruction files, which were generated in the (In Situ) Normalization Calculation command, to execute the transfer of volumes. These files are protected using a hash value, editing them will cause the script execution to fail.

Path to Pipetting CSV File: If the output from the normalization command shall not be used, or you wish to edit the pipetting instruction files before the transfer of volumes, a CSV file can be selected on the computer and used to transfer the liquids.

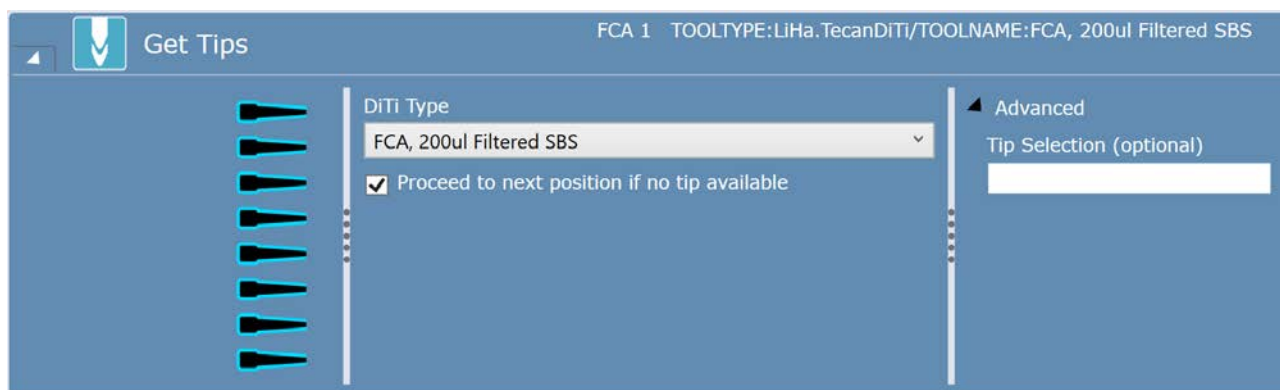
Source Labware: This field contains the name of the labware that is used to aspirate the liquid from. If the output from the Normalization command is used, this must be the same labware as the Sample or Diluent labware, depending on which you wish to transfer. Either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Target Labware: This field contains the name of the labware which is used to dispense the liquid to. Either type your expression or click the labware displayed in the Worktable editor window (Worktable Click).

Liquid Class: Choose the required liquid class from the dropdown list or by means of an expression typed in (e.g., by using a variable defined within your script). The list only shows liquid classes that are suitable for the current configuration of the pipetting instrument.

Advanced Parameters: The Advanced parameters contain the same functionality as the Sample Transfer and Reagent Distribution Smart Commands. For more information on these, refer to Smart Commands.

11.2.5 Get Tips



This command picks up disposable tips (DiTis) of the specified type from a DiTi rack. The application keeps track of the available tip positions on the worktable and automatically picks up the next available unused DiTis of the chosen type.

This information is also persisted when shutting down the application or powercycling the instrument/PC, and is used when executing a run with “stored tip positions”.

Order of Tip Pickup

If there is more than one tip box containing tips of the selected type, pickup starts at the box with the lowest index number (e.g., FCA_200ul[001] before FCA_200ul[002]). In general tips are picked up from left to right within a box, but preference is given to fast pickup over strict left-to-right use—e.g.: 8 tips will always be picked up from a full column and not in two steps starting from an incomplete column in a tip box.

DiTi Type

Click the arrow in the dropdown list to see more options and select the DiTi type of the FCA arm from the list. The tip type can also be selected by clicking a tip box on the worktable.

Proceed to next position if no tip available

If no DiTi is present at the current position and this option is selected, the software automatically moves to the next position. If a tip is detected but cannot be successfully picked up an error message is shown. Follow the instructions of the error message.

Advanced Parameters

Tip Selection (optional):

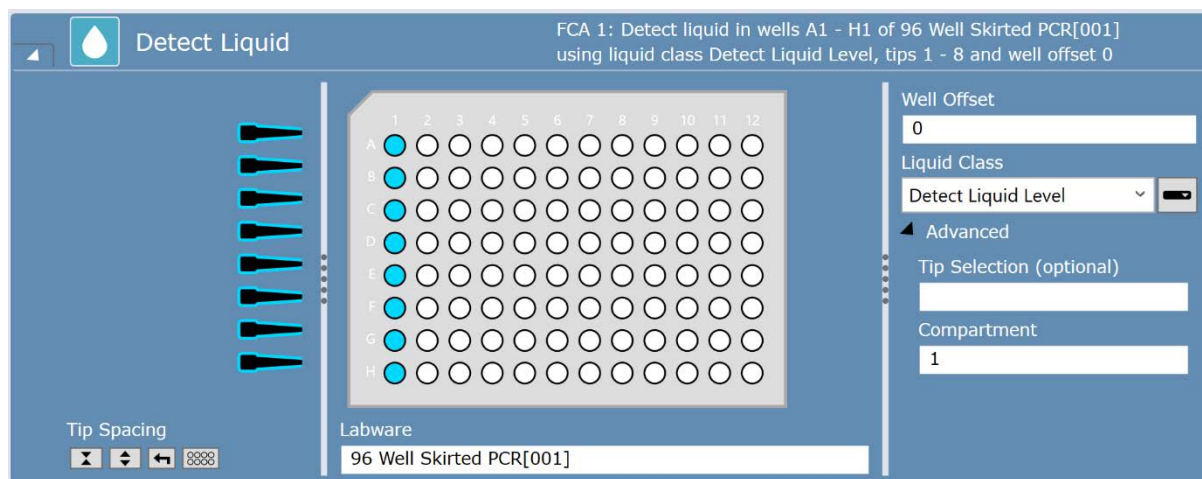
The Tip Selection (optional) parameter is resolved as integer value (integer variables and expressions are allowed) and then interpreted as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Also see about this

["FCA" ▶ 176](#)

11.2.6 Detect Liquid



This command is used to carry out liquid level detection without pipetting and reports the liquid volume for each of the selected wells. The detected liquid volume is directly written into the corresponding well attribute and can later be read out via the `GetAttribute()` function.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support standalone liquid level detection.

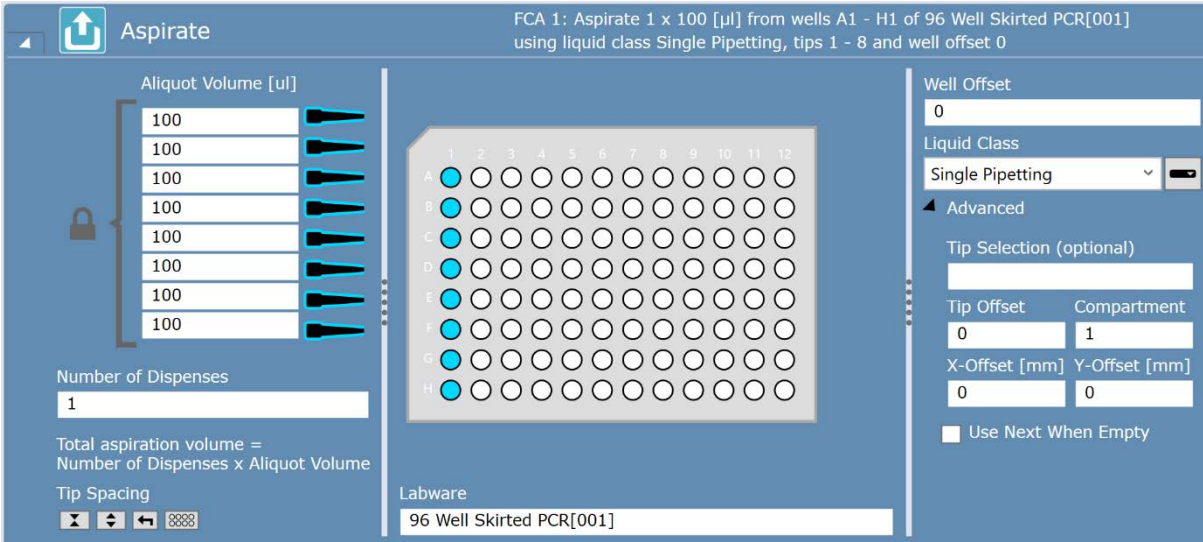
Advanced Parameters

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

11.2.7 Aspirate



Aspirate

FCA 1: Aspirate 1 x 100 [ul] from wells A1 - H1 of 96 Well Skirted PCR[001] using liquid class Single Pipetting, tips 1 - 8 and well offset 0

Aliquot Volume [ul]

100
100
100
100
100
100
100
100

Number of Dispenses
1

Total aspiration volume =
Number of Dispenses x Aliquot Volume

Tip Spacing
[X] [Y] [Z] [W]

Labware
96 Well Skirted PCR[001]

Well Offset
0

Liquid Class
Single Pipetting

Advanced

Tip Selection (optional)
[]

Tip Offset
0

Compartment
1

X-Offset [mm]
0

Y-Offset [mm]
0

☐ Use Next When Empty

This command aspirates liquid from specified positions on the worktable.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Enter the aliquot volume to pipette. If the padlock icon is opened, individual volumes can be selected; when closed, all selected channels will use the same volume. To open the padlock, click on it.

Enter the number of dispenses that are going to be performed after this aspiration. The FCA will then aspirate the required total volume according to the number of dispenses and the aliquot volume.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.

- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- It is not possible to select more wells than the number of tips selected.
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support aspiration.

Advanced Parameters

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer value (integer variables and expressions are allowed) and then interpreted as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

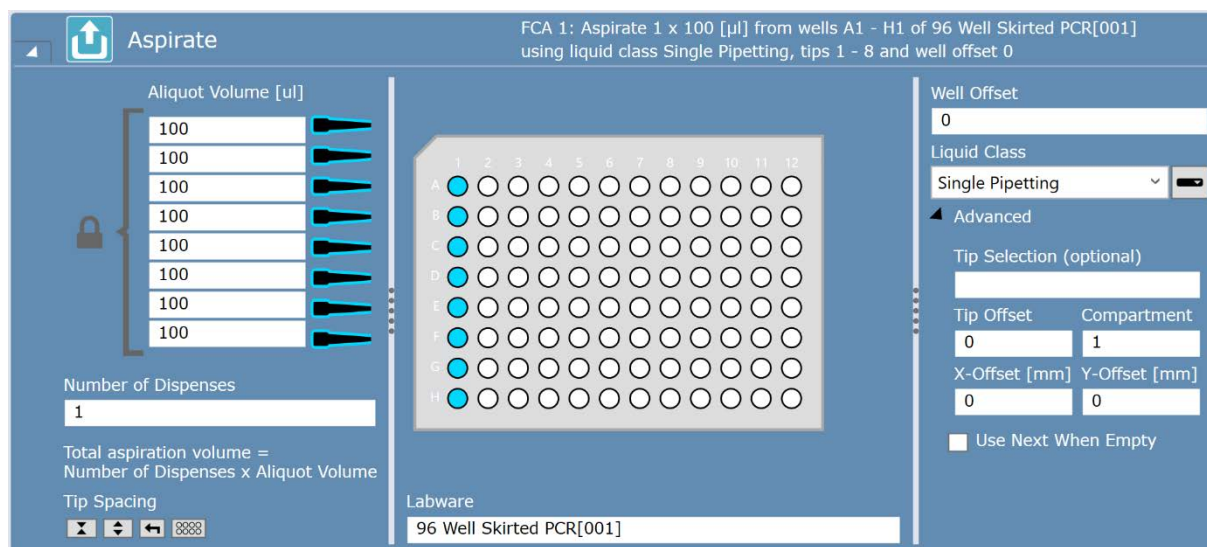
X offset [mm]: This is the offset of the tip from the center of the target well (negative = to left, positive = to right).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Use next when empty: When a liquid-detection error occurs (i.e., no liquid/not enough liquid), the FCA will use the next available labware with the same name and next index number—e.g.:

When "Trough[001]" is empty, the FCA will log the detection error, but it will not display a user prompt. It will immediately go to the labware named "Trough[002]" and try to detect liquid at that location. If all labware items in the group have been emptied, the software will prompt the user to refill them and start with the first selected labware again.

11.2.8 Dispense



Aspirate

FCA 1: Aspirate 1 x 100 [ul] from wells A1 - H1 of 96 Well Skirted PCR[001] using liquid class Single Pipetting, tips 1 - 8 and well offset 0

Aliquot Volume [ul]

100

100

100

100

100

100

100

100

Number of Dispenses

1

Total aspiration volume =
Number of Dispenses x Aliquot Volume

Tip Spacing

8mm

Labware

96 Well Skirted PCR[001]

Well Offset

0

Liquid Class

Single Pipetting

Advanced

Tip Selection (optional)

All

Tip Offset

0

Compartment

1

X-Offset [mm]

0

Y-Offset [mm]

0

☒ Use Next When Empty

This command dispenses a given amount of liquid to specified positions on the worktable.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Enter the volume to pipette. If the padlock icon is opened, individual volumes can be selected; when closed, all selected channels will use the same volume. To open the padlock, click on it.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Advanced Parameters

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support dispensing.

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = to left, positive = to right).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Use next when full: If this option is selected, the software performs the following checks before executing the dispense:

If the selected labware contains the custom attribute *LiquidWasteCapacity*, the software checks whether the remaining capacity (i.e. the *LiquidWasteCapacity* minus the current volume in the cavity) is sufficient for the volume to be dispensed. If the remaining capacity is insufficient, the software will automatically switch to the next labware in the group (e.g. if "Trough[001]" is full, it will choose "Trough[002]"). If all labware items in the group are full, the software will prompt the user to empty them and start with the first selected labware again.

11.2.9 Mix

Mix

FCA 1: Perform 2 cycles of mixing with 50 [µl] in wells A1 - H1 of 96 Well Skirted PCR[001] using liquid class Mixing, tips 1 - 8 and well offset 0

Well Offset: 0

Liquid Class: Mixing

Cycles: 2

Advanced

Tip Selection (optional):

Tip Offset: 0 Compartment: 1

X-Offset [mm]: 0 Y-Offset [mm]: 0

Tip Spacing: [Icons]

Labware: 96 Well Skirted PCR[001]

This command mixes the liquid(s) at specified positions on the worktable.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Enter the volume to pipette. If the padlock icon is opened, individual volumes can be selected; when closed, all selected channels will use the same volume. To open the padlock, click on it.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- Clicking on a selected well in the labware control selects it for one additional tip.
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support mixing.

Cycles: The number of mixing cycles to be executed.

Advanced Parameters

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent

and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

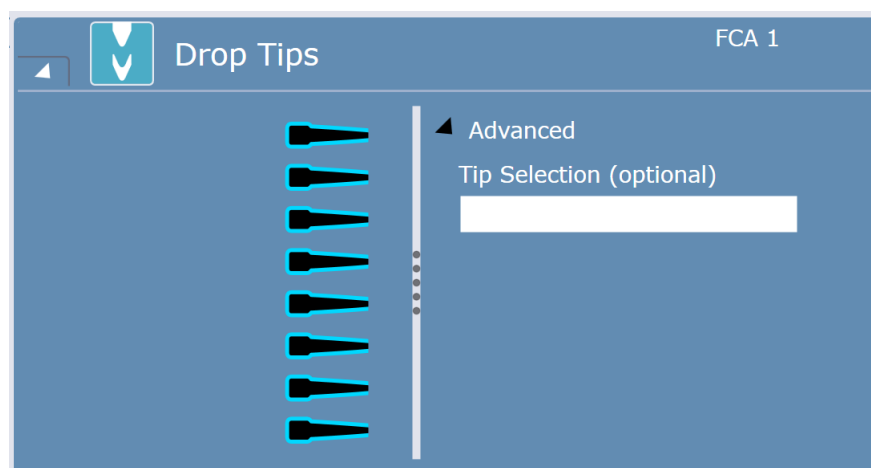
X offset [mm]: This is the offset of the tip from the center of the target well (negative = to left, positive = to right).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Also see about this

["FCA" ▶ 176](#)

11.2.10 Drop Tips



This command discards disposable tips (DiTis) from the selected channels into a DiTi Waste.

The software automatically chooses a suitable waste labware on the worktable.

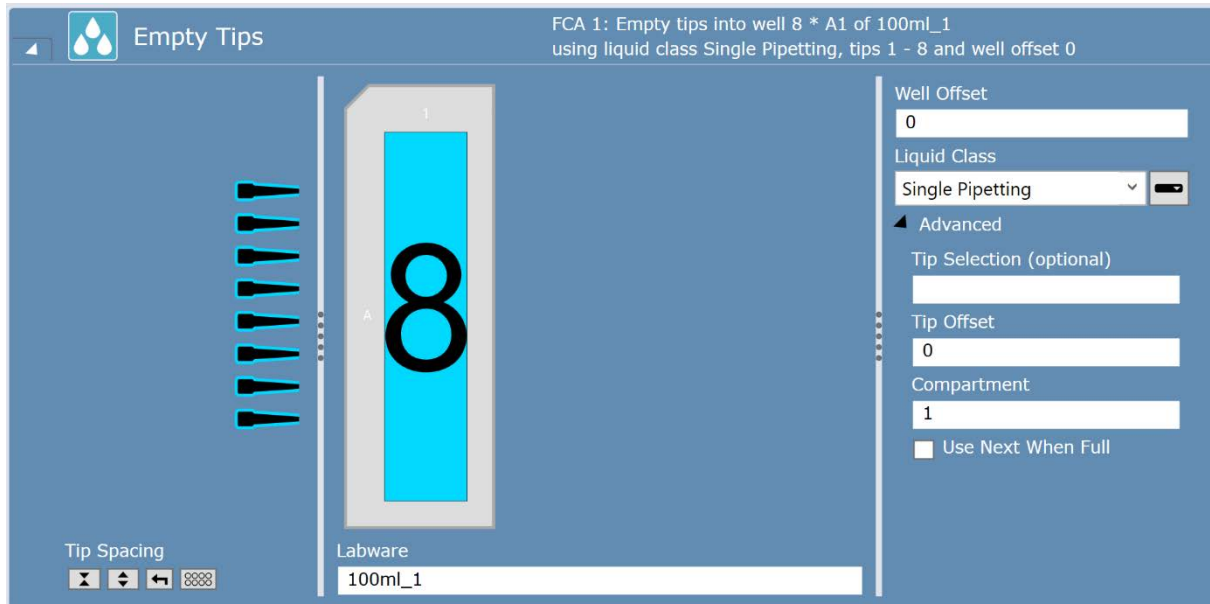
Advanced Parameters

Tip Selection (optional):

The Tip Selection (optional) parameter is resolved as integer value (integer variables and expressions are allowed) and then interpreted as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting

. The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

11.2.11 Empty Tips



This command dispenses the entire content of the selected tips into specified positions on the worktable.

Parameters

Tip Selection:

Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Tip Spacing: The Tip Spacing parameter controls the distance between adjacent pipetting tips for this command. Tip spacing is only relevant if you want to use more than one tip. You can only choose values for tip spacing which are meaningful for the labware geometry. The control offers the following options (from left to right):

Contract:

- If wells are already selected in the labware control and the contracted well selection spacing is physically possible, the well selection alters to match the new well selection cursor spacing.
- If wells are already selected in the labware control and the contracted well selection spacing is physically not possible, only the first well is selected for all tips.
- If only one well is selected in the labware control, clicking Contract has no effect on the selected well.

Expand:

- If wells are already selected in the labware control and the expanded selection is physically possible to select, the well selection alters to match the new well selection cursor spacing. The first selected well is not altered by the Tip Spacing buttons.
- If wells are already selected in the labware control and the expanded selection is not physically possible to select, clicking Expand has no effect.

Reset:

- Reset returns the tip spacing to the default.
- If wells are selected, it selects additional wells by basing a default spacing on the selected tips and starting with the first selected well.
- If wells are selected and it is physically not possible to select additional wells by basing a default spacing on the selected tips starting with the first selected well, all selected wells are cleared.

Independent Selection:

- Clicking on an unselected well in the labware control selects it for one tip.
- Clicking on a selected well in the labware control selects it for one additional tip.
- It is not possible to select more wells than the number of tips selected.
- If there are the same number of wells selected as tips and you click on a selected well, it reduces the number of tips in the well by 1 (potentially unselecting the well).
- The tip to well assignment is consecutive where tips are assigned from 1 to n to the corresponding 1 to m wells according to the well selection per tip.

If the number of tips selected is lowered to a number smaller than the number of selected wells, a corresponding number of selected wells switches to not selected starting with the highest numbered well.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Liquid Class: Choose the required liquid class from the pulldown list or by means of an expression typed in (e.g. by using a variable defined within your script). The list only shows liquid classes that are suitable for the pipetting arm and which support the empty tips workflow.

Advanced Parameters

Tip Selection (optional): The Tip Selection (optional) parameter is resolved as integer (integer variables and expressions are allowed) and interpreted then as a bitmask. The tip selection in the script command is calculated considering both the Tip Selection (optional) bitmask and the tips selected via GUI (binary AND). Thus, only tips selected in both masks will be used for pipetting.

The Tip Selection (optional) mask is specified as a numeric, bit-coded value (tip1 = 1, tip2 = 2, tip3 = 4, tip4 = 8, tip5 = 16, tip6 = 32, tip7 = 64, tip8 = 128). Use the sum of the bit-coded values to select more than one tip. In the GUI, tip1 is the topmost tip.

Tip Offset: This parameter lets you modify the tip selection for the pipetting command. The default is 0, which deactivates the offset. A positive value increments the tip number(s) which are specified in the pipetting command. For example, if the pipetting command specifies tips 2 and 3 and you set this parameter to 1, tips 3 and 4 will be used instead. Negative values of the parameter shift the tip selection in the opposite direction.

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent

and are often smaller in size. Choose the compartment which you want to use for pipetting (the main compartment is normally compartment 1). Well compartments are specified in the labware configuration.

X offset [mm]: This is the offset of the tip from the center of the target well (negative = to left, positive = to right).

Y offset [mm]: This is the offset of the tip from the center of the target well (negative = to front, positive = to back).

Use next when full: If this option is selected, the software performs the following checks before executing the dispense:

If the selected labware contains the custom attribute LiquidWasteCapacity, the software checks whether the remaining capacity (i.e. the LiquidWasteCapacity minus the current volume in the cavity) is sufficient for the volume to be dispensed. If the remaining capacity is insufficient, the software will automatically switch to the next labware in the group (e.g. if "Trough[001]" is full, it will choose "Trough[002]"). If all labware items in the group are full, the software will prompt the user to empty them and start with the first selected labware again.

11.2.12 LLD Error Handling (FCA)

The **LLD Error Handling** command allows to set up default handling options for liquid handling. These options override the settings specified in the liquid class and are valid until the current script ends.



Fig. 89: LLD Error Handling

To disable LLD Error Handling for a part of the script, insert a second **LLD Error Handling** command, and select **User Prompt** or **Liquid Class Defaults** in the **Error Handling Action** box.

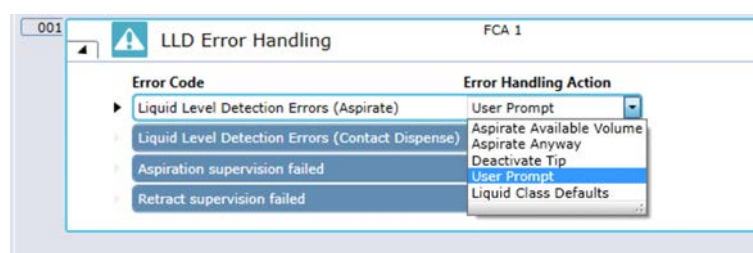


Fig. 90: Error Handling Action box

11.2.13 Variable Pipetting Loop (FCA)

This command allows the automated, iterative execution of a set of script commands – representing a pipetting sequence – with automated variable parameterization.

The system prevents by context check that the user groups the variable pipetting loop command within itself. It is allowed to nest the variable pipetting loop group command inside other group and loop commands.

Parameters

Description: Describe the variable pipetting loop.

Loop Properties

Name of Loop Variable: Define the variable counting the loop cycles. Use the Declare button to register a variable, as defined in this parameter, with its name and type under the script section Variables and with read/write permission.

Sample Count: Stores the expression for the number of samples to pipette. Use the Declare button to register a variable, as defined in this parameter, with its name and type under the script section Variables and with read/write permission.

Default Increments per Cycle

Aspirate Well Offset: Define the well offset for Aspirate commands.

Dispense Well Offset: Define the well offset for Dispense commands.

Update Grouped Commands

Apply: After the group command parameters are set by the user and as each script command is added to the group or when the Apply button is pressed, the system performs automated parametrization for the script parameters Tip Mask and Well Offset for all grouped script commands as follows:

1. For the script parameter Tip Mask, the system uses the expression “tip-sNeeded(tipMask, sampleCount)”. You can edit this parameter after it has been automatically set.
2. For script parameter Well Offset:
 - a. If you select an automatic parametrization for Aspirate, the system uses the expression “aspirateCount() * (<Aspirate Aspiration Offset>)” for every Aspirate within the Variable Pipetting Loop. You can edit this parameter after it has been automatically set.
 - b. If you select an automatic parametrization for Dispense, the system uses the expression “dispenseCount() * (<Dispense Dispense Offset>)” for every Dispense within the Variable Pipetting Loop. You can edit this parameter after it has been automatically set.

If the Variable Pipetting Loop command is executed, the system repeats the sequence of script commands within the loop command until the specified number of samples has been pipetted, i.e. the sampleCount() function equals zero at end of the sequence.

System Functions

sampleCount(): This function returns the number of samples remaining to be pipetted. The initial value is determined from the parameter "Sample Count". At the end of each Variable Pipetting Loop, the value decrements by the number of distinct tips having aspirated.

aspirateCount(): This function returns the number of aspirate commands having been executed within the Variable Pipetting Loop.

dispenseCount(): This function returns the number of dispense commands having been executed within the Variable Pipetting Loop.

11.2.14 Move Arm

This command moves the FCA to a position over a selected labware without performing any pipetting operation.

Parameters

Tip Selection: Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

Labware: In the Labware box, either type your expression or click the labware displayed in the Worktable editor window (Worktable click).

Movement Speed [mm/s]: Select the speed to use for the movement in millimeters per second.

Well Offset: Select by how many wells the starting point of the currently selected wells is shifted. This is usually expressed as a variable.

Z-Position after movement: Select the target Z-position of the arm. Range: ["Z-Travel", "Z-Start", "Z-Dispense", "Z-Max", "Z-Bottom"]. Default: "Z-Travel". The software will first move the arm to Z-Travel of the labware by a pathfinder controlled movement and then move the selected tips to the Z-Position.

Z-Offset [mm]: Defines an offset in millimeters to the Z-Position after movement (positive = lower, negative = higher).

Advanced Parameters

Compartment: Some special labware types (e.g., for protein crystallography) have wells with several compartments (cavities). Each well typically has one main compartment and one or more satellite compartments which are directly adjacent and are often smaller in size. Choose the compartment which you want to use as target for the move.

11.2.15 Start Prepare Pressure

This command starts the preparation of the pressure tanks of the FCA for a subsequent pipetting workflow. Since pressure preparation may consume a certain amount of time, this command can be used in advance of the actual pipetting, so that the system can build up the pressure while still being occupied with other tasks.

Parameters

DiTi Type: The tip type which is planned to be used for the subsequent pipetting workflow.

Liquid Class: The liquid class intended to be used for the subsequent pipetting workflow.

11.2.16 Channel Self Test

This command performs a test of the inline filter of the FCA channels to detect potentially damaged filters.

In order to protect the pipetting channels against over-aspiration of liquid (e.g. if a disposable tip with smaller capacity than intended is used) a control system is installed on each channel. The control system combines a filter inside the DiTi cone and the pressure monitoring of the channel.

This command can only be executed without mounted disposable tips.

Parameters

Tip Selection: Select the tips you want to use. By default, the software selects all tips. To turn off or turn on a tip, click on it. If the box is framed in red, click to select a tip.

The software automatically executes a channel self-test in case of a pressure profile error during pipetting. In that case, the self-test is performed as part of the Drop Tips command, after the tips have been discarded from the affected channels.

If the self-test fails, the affected channel is deactivated for further usage (marked as “broken” in the configuration).

11.2.17 Pressure Monitored Pipetting (PMP)



Currently the PMP functionality is limited to the evaluation of liquid aspiration.

How to activate the PMP feature

The availability of the PMP feature depends on a corresponding license being available for the connected instrument. This is checked during startup of the application.

How to set up a liquid class for PMP

If the PMP feature is activated, the liquid class editor provides additional settings in the Aspirate section:

- Error handling option in case of an aspiration error
- Evaluation model
- Threshold values depending on selected evaluation model



Currently only the following tip types and volume ranges are supported:

- 200 μ l DiTis (filtered and non-filtered): 20 μ l – max.
- 350 μ l DiTis: 20 μ l – max.
- 1000 μ l DiTis (filtered and non-filtered): 50 μ l – max.
- 1000 μ l Wide-Bore DiTis: 100 μ l – max.

Threshold values for neural nets

If a neural net for classification has been selected as evaluation model, the user can define the following parameters to control sensitivity. The neural net determines 2 probabilities for each pipetting curve:

- The probability that the pipetting is correct
- The probability that the pipetting has an error

where 1 represents 100% certainty and zero represents no certainty. The user can control the sensitivity of the evaluation by defining how certain they want to be for a Correct or Error determination.

Parameter	Minimum	Maximum	Description
Correct Limit	0.0	1.0	If the “Correct” probability is equal to or greater than this limit, the classification is reported as “Correct” even if there is an error rating with higher probability.

Error Limit	0.0	1.0	If the highest probability of an error rating does not surpass this limit, the classification is reported as "Correct".
-------------	-----	-----	---

If a neural net for anomaly detection has been selected as evaluation model, the user can define the following parameters to control sensitivity.

Parameter	Minimum	Maximum	Description
warnThreshold	0.0	10.0	If the anomaly value (deviation from ideal pressure curve) exceeds this threshold, a warning is raised and the sample is flagged accordingly, but the run continues. There is no error handling being performed.
errorThreshold	0.0	10.0	If the anomaly value exceeds this threshold, this is treated as a PMP Aspiration Error and the selected error handling is executed.

Error handling options

The following error handling options can be configured in the liquid class to be performed in case of an aspiration error being reported by PMP:

User Prompt: A prompt will be shown, requiring the operator to choose the actual error handling option at runtime. The operator can choose one of the following options.

Ignore and Continue: The run continues, but a warning message is shown in the System Trace View.

Dispense back to source and deactivate tip: Liquid is dispensed back to the source well, the trailing air gap is aspirated and the channel is deactivated until the end of the current pipetting sequence.

Dispense to waste and deactivate tip: After the non-affected channels have completed the aspiration, the FCA moves to the liquid waste, dispenses the liquid and deactivates the channel until the end of the current pipetting sequence.

Retry once with new tip: Liquid is dispensed back to the source well. After the non-affected channels have completed the aspiration, the FCA mounts new DiTis on the affected channels and retries the aspiration once. If the retry also fails, the error handling reverts to “Dispense back to source and deactivate tip”.

How to use the PMP feature in a script

If a PMP license is available for the connected instrument, the software provides two additional script commands for the FCA:

- Switch On PMP
- Switch Off PMP

The **Switch On PMP** command activates pressure data streaming for all channels. This means that a pressure curve will be recorded for each pipetting sequence, starting with the pickup of a disposable tip and concluding with its being discarded, until the channels are deactivated by a Switch Off PMP command.

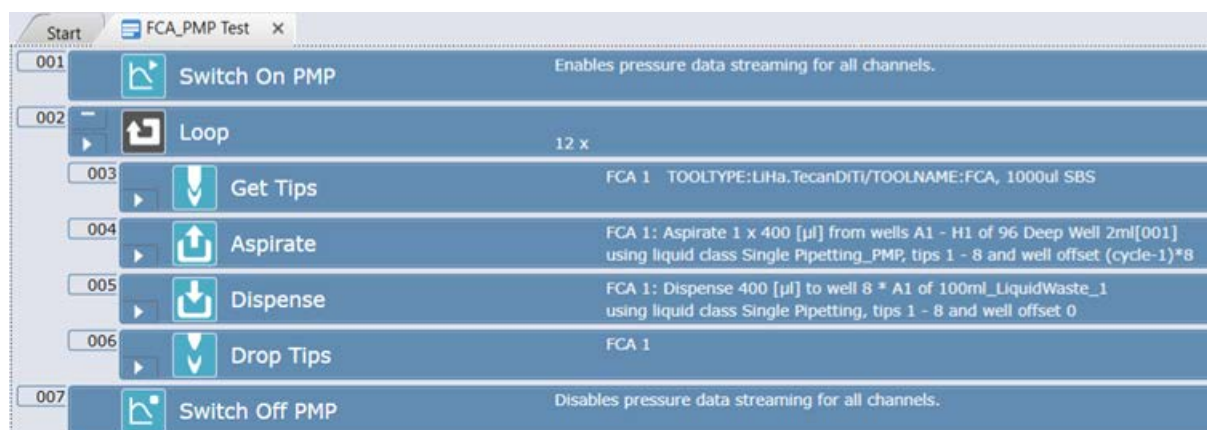
The curve recording and evaluation will also be performed for worklists and Smart Commands while PMP is switched on.

The recorded pressure curves are stored in:

C:
\\ProgramData\\Tecan\\VisionX\\MapDataBase\\MAP.ArcPipettingData\\Measurement

In addition to the curve recording, the aspiration of liquid will be evaluated according to the settings in the liquid class (PMP evaluation model).

The **Switch Off PMP** command deactivates pressure data streaming for all channels.



Sample Tracking

The User can integrate sample tracking into scripts using PMP. Samples must be registered and have unique barcodes. Reports can be generated by selecting a PMP Plate Map Report which will include the sample ID, the pipetting status, and a link to the curve file.

11.2.18 Pipetting System Check

This command performs a test procedure to check the functionality of the pipetting system. The test requires that a Tightness Test Block labware is placed on the worktable, at a position reachable for the pipetting arm. This command can only be executed without mounted disposable tips.

Parameters

The command does not have any parameters. The test is always executed on all channels of the arm.



Pipetting System Check

FCA 1 Executes pipetting system check on all channels.

If the pipetting system check fails, the run will finish with errors. If the application is running under a Gx license, the check has to be executed at least once in any run that uses the pipetting arm. In case the check is not executed, the run will finish with warnings.

11.3 FCA Gripper

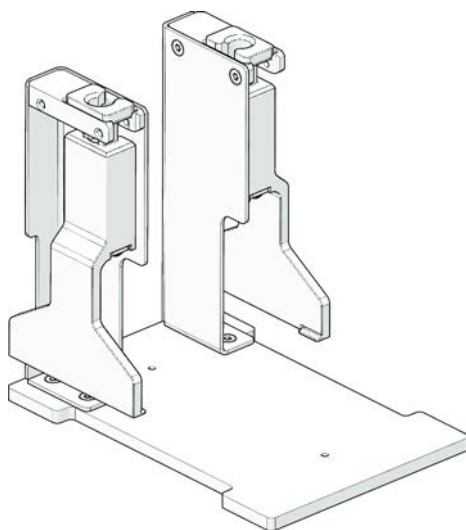


Fig. 91: FCA gripper

The FCA gripper is an option for the FCA that—in addition to pipetting—also allows using the arm for moving labware. Instead of disposable tips FCA gripper fingers are attached to the FCA. The FCA can automatically get and drop the FCA gripper fingers during the run. The FCA gripper allows moving labware (plates and DiTi boxes) with the FCA Transfer Labware command—as an alternative to the RGA Transfer Labware command.

FCA Gripper Fingers

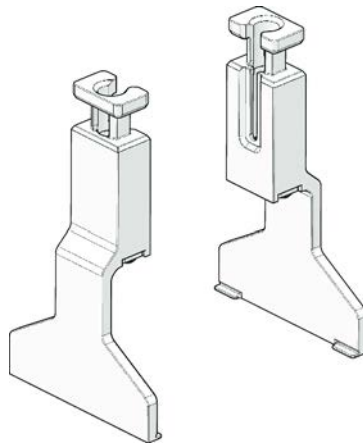


Fig. 92: FCA gripper fingers

The FCA can get and drop the FCA gripper fingers with the Get Fingers and Drop Fingers commands.

The FCA gripper fingers are designed for up to 20,000 pickup moves. When 90% of this limit has been reached, a different warning message that the fingers need to be replaced will be shown. A new warning message is shown when 100% of the lifetime is reached.

Once the fingers have been replaced, a key operator can confirm the replacement and reset the counter by clicking **Settings > Configure System > Devices > FCA > Reset Counter**.

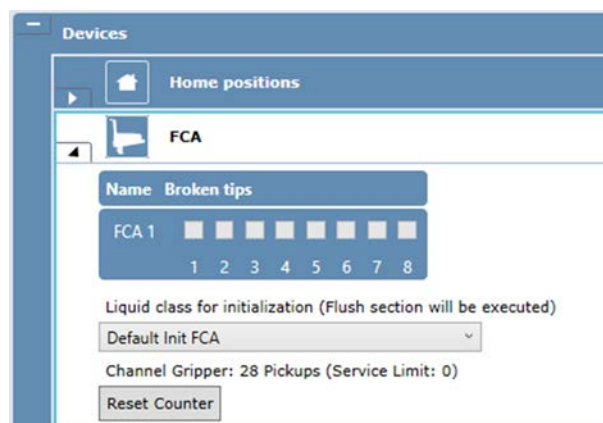


Fig. 93: Reset Counter

FCA Gripper Docking Station

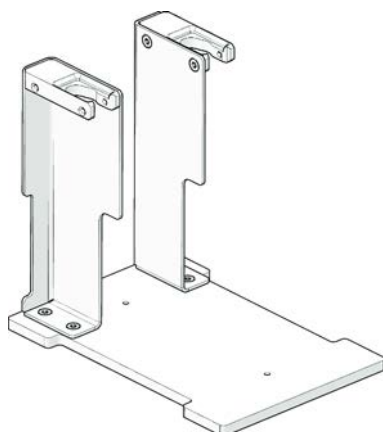


Fig. 94: FCA gripper docking station

The FCA gripper docking station is a docking station for the FCA gripper fingers that is mounted on a standard microplate nest position on the deck of the Veya. At runtime the FCA gets the FCA gripper fingers from this position with the Get Fingers command and places them back to this position with the Drop Fingers command.

The FCA gripper docking station consists of two upright holders. The distance between them can be specified in the **Gripperwidth** custom attribute of the gripper nest.

11.3.1 Transfer Labware

Transfer Labware		FCA 1 Transfer '96 Well Skirted PCR[001]' to 'Nest7mm_Pos' position '10'	
<input type="radio"/> Back to Base <input checked="" type="radio"/> To Location	Labware 96 Well Skirted PCR[001]	Target Location Nest7mm_Pos	Target Position 10
			<input checked="" type="checkbox"/> Only use selected site

This command moves labware on the deck of the instrument using the Channel Gripper Fingers. If there are no fingers mounted, the software will automatically mount them on a suitable channel pair.

Parameters

Back to Base: The system will move the labware to its base location. If this option is selected, the UI controls for location and site are grayed out.

To location: The system will move the labware to the specified location.

Labware: Specify the labware to be transferred by the FCA by entering the labware name (or an expression that resolves into a labware name) or by clicking the labware displayed in the Worktable editor window (Worktable click).

Target location: Select a target location by entering the location name or by clicking the target site on the worktable.

Target Position: By default the system will place the labware on the first free site of the target location. The user can override this behavior by checking Only use selected site and specifying a dedicated site on the target location. The target site can be specified by entering the site number (or an expression which resolves into a site number) or by selecting a carrier or site on the worktable.



Presence Check The FCA performs a labware presence check both before picking up the labware at its source position and after placing it on the target position.

Move Simulation The software simulates the move in Move Simulation mode (Script Editor) and updates the worktable if the move is possible.

Grip Position By default the labware is gripped at the bottom. This behavior can be changed by means of the custom attribute GrippingZOffset, which specifies a Z-Offset in mm that is applied when gripping the labware. Such a grip offset is recommended when moving SBS DiTi boxes (the default DiTi labware definitions already include that custom attribute).

11.3.2 Get Fingers

This command picks up the Channel Gripper Fingers from a gripper station on the worktable.

Parameters

Gripper Fingers: Select the gripper fingers to be picked up, either by typing the label or by selecting the grippers by clicking them on the worktable.

Mounting Channels: Select the channels on which to mount the gripper fingers. Depending on the arm configuration, the following channels will be used (1 = rearmost channel):

Mounting Channels	8-Channel FCA	8-Channel FCA
Front	3 and 4	7 and 8
Back	1 and 2	1 and 2
Inner	2 and 3	3 and 6
Outer	1 and 4	Not supported

The Channel Gripper Fingers allow using the arm for moving labware in addition to pipetting. The gripper fingers allow moving plates and SBS DiTi boxes with the *Transfer Labware* command.

11.3.3 Drop Fingers



This command returns the mounted gripper fingers to an empty gripper station on the worktable.

The software automatically selects a suitable empty gripper station.

11.4 RGA

11.4.1 Driver Configuration (RGA)

If the instrument is equipped with an RGA, its general settings can be configured under Configure System|Drivers|RGA ...

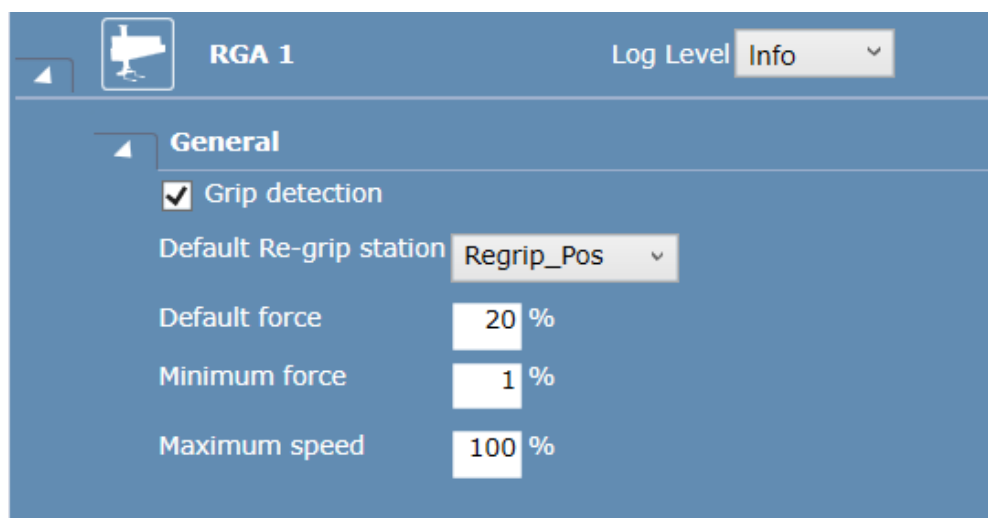


Fig. 95: RGA Driver Configuration

Grip detection: If enabled, the system will check for each plate gripping action whether or not an object was detected by the hardware and trigger the appropriate error handling. If this option is disabled, the system assumes that the grip action is always successful.

Default Re-grip station: Here, the location name of the default re-grip station can be specified. This re-grip station will be used whenever re-gripping is needed for a labware transfer (see Introduction|Regrip) and no re-grip station is specified by any involved location or the labware.

Default force: The default gripping force (in percent) to be used when gripping force is not limited by the labware via the Force custom attributes.

Minimum force: The lower limit for gripping force reduction. This setting can prevent that a too low gripping force is used. A too low gripping force can lead to erroneous object detection and loss of object while moving.

Maximum speed: The upper limit for the movement speed for plate transfers.

11.4.2 RGA Execute Single Vector

Use this command to execute a vector (list of 3D waypoints) with the RGA gripper.

Fig. 96: Executive Single Vector

Move Arm

This option allows the user to specify the movement direction of the selected vector which will be executed. The following directions can be selected:

- From safe position to end position (with the possibility to move back to the safe position)
- From end position to safe position

If the option **From safe position to end position** is selected, the arm will execute the movement vector like it is defined in the selected Location (from top to bottom). Selecting the option **and back to safe position** will execute the vector again in reverse order (e.g. to grip an object and move out again).

If the option **From end position to safe position** is selected, the arm will only execute the movement vector in reverse order.

In addition to the execution of the movement vector, the grippers can be opened and closed. This can be done before and/or after the move. The following options can be selected:

Before Move

This option will open or close the gripper to a specified width, in millimeters, before the move vector is executed. The **Close** option will use grip detection (if it is enabled in the Configure System menu) to detect a labware item in between the grippers. If the gripper reaches the specified position without detecting a labware item, an error is raised. The **Open** option is executed without grip detection.

After Move

This option will open or close the gripper to a specified width, in millimeters, after the move vector is executed. The **Close** option will enable a grip detection to detect a labware item in between the grippers. If the gripper reaches the specified position without detecting a labware item, an error is raised. The **Open** option is executed without grip detection. By default, neither **Before Move** nor **After Move** is selected. This setup which will only execute the selected movement vector without moving the gripper.

Parameters

Location and Position: Specify the name of a location and a site number for which the vector shall be executed. This can be done by clicking a carrier, site or labware on the worktable.

Vector: Select a vector name from the **Vector box**. The list contains the names of all vectors for the current robot stored in the selected carrier.

Speed: Specify the move speed in percent (1–100%).

Grip force: Specify the gripper force in percent (1–100%).

All edit fields support variables and expressions.

11.4.3 Transfer Labware (RGA)

Use this command together with an RGA to transfer labware objects or labware-like carriers (i.e., less than 400 g) to a location.

Carriers

When transferring carriers, the carrier name has to be typed in manually. The carrier needs CustomAttribute **MoveableCarrier** set to true and to specify CustomAttributes **Grip_Narrow** and **Grip_Wide**.

24 Eppendorf Adapter is a moveable carrier and can be used as example.



Pay close attention to copy and pasting lines before and after Transfer Labware commands. For details, see the Labware rotation and worktable simulation topic.

You can specify the labware to be transferred by the RGA. This can be done by entering the labware name or an expression which resolves into a labware name into the **Labware** box or by selecting labware on the worktable.

You can select from the following, mutual exclusive options:

- **Move to location:** You can specify the location the labware will be transferred to.
 - By default the system will place the labware on the first free site of the target location. The user can override this behavior with a check box and specify a dedicated site on the target location. The target site can be specified by entering the site number or an expression which resolves into a site number into a text box or by selecting a carrier or site on the worktable.
- **Move to base:** The system will move the labware to its base. If this option is selected, the UI controls for location and site are dimmed.
 - By default the system will place the labware on the first free site of the target location. The user can override this behavior with a check box and specify a dedicated site on the target location.

Context check

The system will check if the move is possible (labware exists, location/target site is available) and raise context check messages if an error is detected.

Move Simulation

The system will simulate the move in Move Simulation mode and update the worktable (move the labware to the destination) if the move is possible.

Example

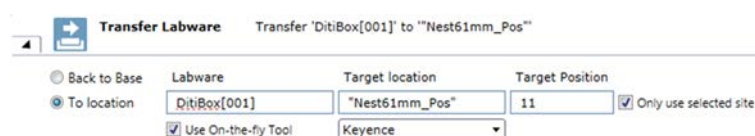


Fig. 97: Transfer Labware

Parameters

Labware: In the Labware box, either type your expression or click the labware displayed in the **Worktable** editor window (Worktable Click).

To location: Click **To location** to specify the location the labware should be moved to. In the **Target location** box, type the location name (or an expression which resolves into a location name) or select a carrier or site on the worktable.

Back to Base: Select **Back to Base**, to move the labware to its base—i.e., the original position where the labware was placed in the initial workspace. If the **Back to Base** option is selected, **Target location** is set to "<BASE>" and appears dimmed. **Target Position** appears dimmed also.

Target location: Select a target location for used DiTi racks in the list box. The usage of "<BASE>" as an alias for the base location is supported.

Target Position: The default is to place the labware on the first free site of the target location.

Only use selected site: To change the default, select the **Only use selected site** check box and specify a dedicated site on the target location.

Target Position

☒ Only use selected site

Fig. 98: Target Position

To specify the target site, click in the **Target Position** box and type the site number (or an expression which resolves into a site number) or select a carrier or site on the worktable.

Use On-the-fly Tool: Select this check box to use a barcode reader as on-the-fly tool. The barcode reader must be available on the worktable, otherwise this field is not visible.

The vector, which is used to move the labware to the barcode reader, is selected depending on the grip mode that is used for the transfer from the source to the destination of the transfer.

There is also the possibility to move the RGA while the barcode scanner is scanning the labware. In order to activate this functionality an additional vector must be created in the carrier of the barcode scanner.

The name of these custom vectors must begin with "OnTheFly" and contain the grip mode for which it shall be used (e.g., "OnTheFlyNarrow"). The first point of such a vector should be identical with the end position of the normal vector for the barcode scanner. When the scan is executed and the corresponding custom vector is present, the vector is moved from start to end position and back again. The speed of the movement can be reduced via the custom attribute **OnTheFlySpeed**. Here a speed factor can be set (1–100%).

Labware rotation and worktable simulation

vControl tracks labware rotation. It also simulates the worktable as the program is being written. Both of these features make it very easy to see what will happen at run time when the script/method is executed.

Pay close attention to copy and pasting lines before and after **Transfer Labware** commands. If a labware is rotated after the pipetting commands are written, it may impact the well selections. After you expand the inserted command once, the values get updated as expected.

Because vControl is an open system, during method development one must always verify that the intended wells are correctly pipetted. For more information on validation of methods, see “Validation of vControl applications” in Help.

11.4.4 Replace Used DiTi Racks (RGA)

Use this command to replace empty DiTi racks on the worktable with unused DiTi racks.



Fig. 99: Replace used DiTi Racks

Parameters

DiTi Type Filter: Select the disposable tip (DiTi) type to replace with from the list box or type an expression to specify the DiTi type.

Labware Name Filter: Select a labware from the list of all available labware names or by clicking it in the Worktable editor. Alternatively, you can type an expression to specify the labware name filter.

Target location: Select a target location for used DiTi racks in the list box. The usage of **<BASE>** as an alias for the base location is supported.

Only remove: To remove the DiTi rack without replacement, select the **Only remove** check box.

11.4.5 Move Arm (RGA)



Fig. 100: Gripper

Use this command to open or to close the gripper fingers, or make a relative move of the gripper to its current position.

You can select from one of the following options:

Gripper

- **Open gripper to:** Gripper width in mm.
- **Close gripper to:** Gripper width in mm.
- **Force:** Grip force in percent (range: 1–100%).

Movement

- **Move:** Move relative to current position

- **Distance:**
 - Relative distance to move in **A** (X-direction in gripper coordinate system) in mm. Floating point value.
 - Relative distance to move in **B** (Y-direction in gripper coordinate system) in mm. Floating point value.
 - Relative distance to move in **Z** in mm. Floating point value.
- **Speed:** Speed in percent (range: 1–100%). Integer value.

11.4.6 Move Arm (RGA)

Fig. 101: Gripper

Use this command to open or to close the gripper fingers, or make a relative move of the gripper to its current position.

You can select from one of the following options:

- | | |
|-----------------|--|
| Gripper | <ul style="list-style-type: none"> • Open gripper to: Gripper width in mm. • Close gripper to: Gripper width in mm. |
| Movement | <ul style="list-style-type: none"> • Force: Grip force in percent (range: 1–100%). • Move: Move relative to current position • Distance: <ul style="list-style-type: none"> – Relative distance to move in A (X-direction in gripper coordinate system) in mm. Floating point value. – Relative distance to move in B (Y-direction in gripper coordinate system) in mm. Floating point value. – Relative distance to move in Z in mm. Floating point value. • Speed: Speed in percent (range: 1–100%). Integer value. |

11.5 Loading ID for vControl

The system allows loading Tube Runners during a run via the Start Loading command. After the command is finished (operators clicks continue), the barcodes will be associated with the appropriate tubes. The loaded runners are supervised from that point forward and are not allowed to be removed until an Allow Removal command has been used.

Once the IDs are set by completion of the Start Loading command, all the standard vControl barcode handling is available. The ID slots of the tubes can be read via the GetAttribute function. The barcodes can be viewed via the Infopad. The LabwareCount function can return the number of scanned tubes which can in turn be used to impact things like the sample count of a Sample Transfer command.

Worklists can target individual tubes based on barcodes, instead of absolute position. When using worklist with Loading ID Runners, be sure your destination labware is in common range of all eight flexible channel arm tips. This will allow vControl to always find a tip capable of making the transfer.

Using Loading ID to capture tube barcodes followed by Sample Tracking Register Labware commands is a good way to track all barcodes and pipetting actions from the tubes to subsequent labware. It is only necessary to register the runners after the Start Loading command has been used and before the first pipetting step takes place. After all pipetting steps are done, use the Generate Report command and target the destination labware to get an output of all IDs and pipetting actions.



NOTICE

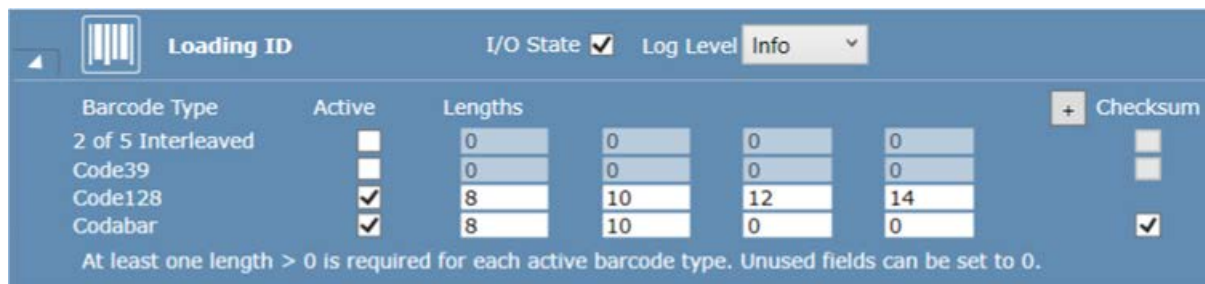
Convention

The following special characters must not be used in barcodes: ! % & ~ () = | as well as carriage return and line feed.

11.5.1 Barcode Types Definition

This configuration page can be used to configure the barcode types which shall be accepted by the Loading ID.

For each active barcode type, up to five lengths can be configured. If a '0' is entered into the length field, the field is ignored. Barcode types and lengths that have not been configured cause an unreadable barcode error when scanned by the Loading ID. It is not possible to configure the scanner to ignore the barcode length.



Barcode Type	Active	Lengths	Checksum
2 of 5 Interleaved	<input type="checkbox"/>	0 0 0 0	<input type="checkbox"/>
Code39	<input type="checkbox"/>	0 0 0 0	<input type="checkbox"/>
Code128	<input checked="" type="checkbox"/>	8 10 12 14	<input type="checkbox"/>
Codabar	<input checked="" type="checkbox"/>	8 10 0 0	<input checked="" type="checkbox"/>

At least one length > 0 is required for each active barcode type. Unused fields can be set to 0.

Code 128 is highly recommended for optimal reading confidence, accuracy and compactness.

On the Settings menu, click Configure System and navigate to Drivers > Loading ID.

11.5.2 Worktable Setup

The segment needs to be loaded onto the worktable. There is one segment for the leftmost (Loading ID Left) end of the instrument and one for any other position of the worktable (Loading ID Middle). There are Loading ID specific tube runners that are needed to load on the segment.

For every connected device a corresponding segment must be placed on the virtual worktable. Every carrier on a virtual worktable must be associated with the driver, refer to section Configuring Your Drivers.

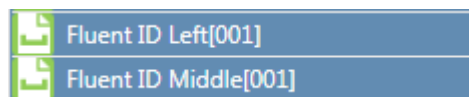


Fig. 102: Connected Loading ID device

A reflector is required to be located on the first grid, right of each segment. Ensure that the worktable layout allows the reflector to be mounted or the reflector sticker to be affixed. All available tube runners, as listed in Veya Operating Manual (refer to section Reference Documents) are available in vControl labware:

	1x24 1.5ml Eppendorf Tube Runner (Runner)
	1x24 10mm Tube Runner no Tubes (Runner)
	1x24 10x100mm Tube Runner (Runner)
	1x24 13mm Tube Runner No Tubes (Runner)
	1x24 13x75mm Tube Runner (Runner)
	1x24 16mm Tubes 100mm (Runner)
	1x24 16mm Tubes Empty (Runner)
	1x24 2ml Eppendorf Tube Runner (Runner)
	1x24 Eppendorf Tube Runner no Tubes (Runner)
	1x26 16mm Tubes 100mm (Runner)
	1x26 16mm Tubes Empty (Runner)
	1x3 320ml Trough Runner (Runner)
	1x3 320ml Trough Runner no Trough (Runner)
	1x32 10mm Tubes 75mm (Runner)
	1x32 10mm Tubes Empty (Runner)
	1x32 13mm Tubes 100mm (Runner)
	1x32 13mm Tubes 75mm (Runner)
	1x32 13mm Tubes Empty (Runner)
	1x32 Safe Lock Tubes 2ml (Runner)
	1x32 Safe Lock Tubes Empty (Runner)
	1x4 100ml Trough Runner (Runner)

Fig. 103: Available Loading ID tube runners

NOTICE

Ensure that the Path Finder uses the right travel positions for the Loading ID

To ensure that Path Finder uses the right travel positions for the Loading ID, populate the virtual worktable with tube runners on every grid where a runner is expected for loading at any time in the run. This way Path Finder will move always in the safe distance, even in case the user placed a runner on a grid outside of a loading time frame.

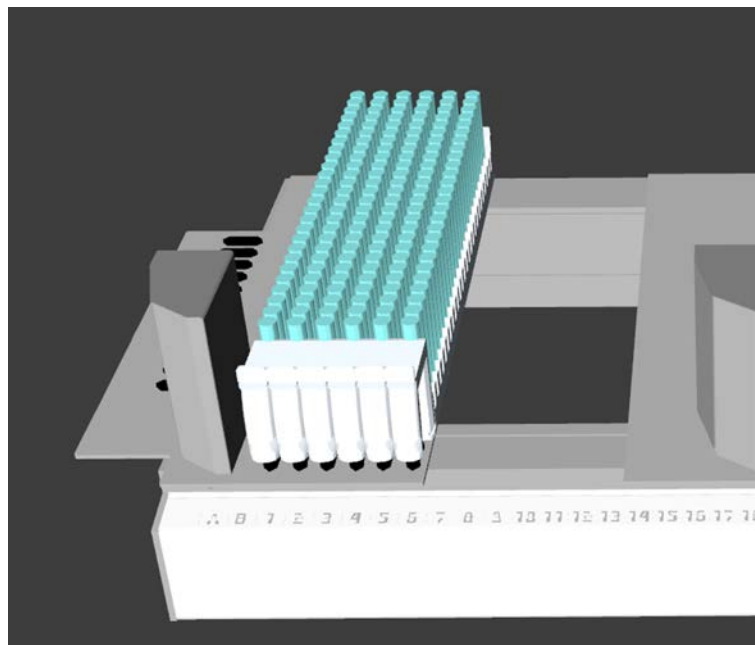


Fig. 104: Virtual worktable layout: runner occupy the 6 grids of the Loading ID segment

The reflector is mandatory to position physically on the worktable. No additional configuration is needed in the software. The bounding boxes of the modules ensure that Path Finder will account for the presence of the reflector at run time.

11.5.3 Start Loading

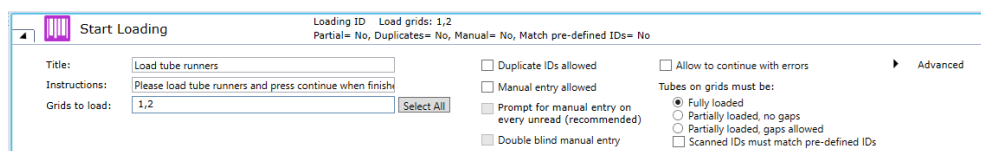


Fig. 105: Start loading command

The **Start Loading** command is used to activate the Loading ID barcode scanner and invite the user to insert tube runners. Multiple Loading ID modules can be used in a single **Start Loading** command. At the point where this command is set in the script, the system requests to load runners onto the Loading ID module for barcode scanning.

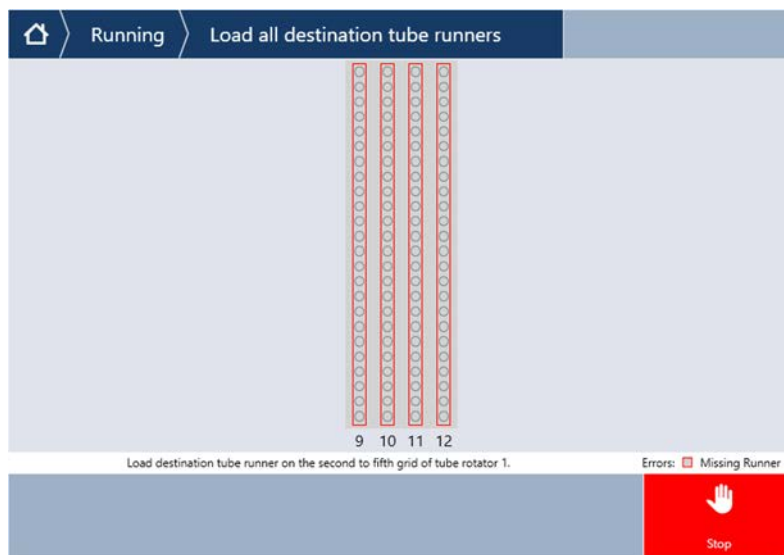


Fig. 106: Load all destination tube runners

When running in simulation mode the command will be skipped and no action is shown to the user.



An Active Stop will occur if a runner is removed from a grid lit in green after the completion on of the Start Loading command. The grid LED will turn red and an error message will be displayed on the touch monitor. If the LED turns red after an unexpected interruption (e.g., runner shifted inadvertently out of its stop position when sliding out a neighbor runner, run-time error dialogue), it must be fully unloaded and reloaded by the operator. The system will verify that the barcodes are the same as read during the initial loading and any mismatch will be displayed on the touch monitor.

*Therefore, it is recommended to set in the script an **Allow Removal** command as soon as the tubes and runners are no longer used for the process (e.g., liquid transfer is completed).*

One part of the **Start Loading** command dialog is for defining the location of Loading ID runners and loading instructions. The second part shows all the options for error handling and expected behavior depending on application.

Title: Define a title for this command. The title is shown during the run on top of the touch monitor user interface.

Instructions: Define the operator instructions shown on touch monitor.

Grids to Load: Define the grids tube barcodes are to be read. Enter ranges or single grid positions into the textbox or click a tube runner on the virtual worktable. Select all, if barcodes are to be read on all available grids. Enter a single grid, individual grids separated by commas, or a range of grids separated by a hyphen (e.g., 1,2,4-6).

Duplicate IDs allowed:



*Do not use the **Duplicate IDs allowed** option with Sample Tracking. Sample Tracking reports are not designed to handle duplicate IDs.*

If this is checked, duplicate IDs on one or more Loading IDs are allowed. If not checked, the operator will get an error message on the touch monitor about duplicate barcodes. Their position will be flagged as errors on the touch monitor.

Manual Entry allowed: When manual barcode entry is checked, the operator is allowed to manually enter barcodes in case of unread barcodes. The touch monitor prompts the user for the unread barcode. The user may enter barcodes directly. The operator may unload the runner. Unread barcodes flagged on the touch monitor can be scanned with a handled barcode reader or typed in. The runner is then loaded again.

Prompt for manual entry on every unread (recommended): If checked, every unread barcode must be manually entered upon loading a runner. If unchecked, the operator will not be required to manually enter a barcode for an unread tube more than once.

Double blind manual entry: For higher security, double blind entry forces the operator to enter the same barcode twice.



When entering barcodes manually, make sure that white spaces are not added by mistake before or after the barcode. Checks for duplicate barcodes take white spaces into account. For example, barcodes " abc123" and "abc123" will not be considered duplicates.

Allow operator to continue with errors

If this box is checked, the run can continue with unresolved errors such as unread barcodes, missing tubes, duplicates, missing runners. LEDs will turn green as soon as the operator continued the run.

Tubes on grid must be:

- **Fully loaded:** Tube and no runner gaps are allowed.
- **Partially loaded, no gaps:** The tube runner may contain more tubes than physical tube runner. The first tube defined in the worktable must be present and no tube or runner are allowed until the last tube physically present in the furthest runner on the right.
- **Partially loaded, gaps allowed:** Tube and runner gaps in runners are allowed.
- **Scanned IDs must match pre-defined IDs:** If this box is checked, the scanned barcodes on the tubes will be verified, if they match the pre-defined barcodes from the barcode attribute of the corresponding tube. Pre-defined barcodes can be set to the tubes via the SetAttribute() function in advance of the scan. Operator is alerted to mis-matching tube-barcodes during loading. If this option is selected, "Allow to continue with errors" is grayed out.

Advanced

Additional settings can be defined in the Advanced tab.

Fig. 107: Advanced settings

Continue script while loading: If checked other steps in the script will run while the operator loads the runners. For every **Start Loading** command a corresponding **Wait for Finish** command must be included right after in the vControl Script.

NOTICE

Allowing the script to continue brings a risk of collision.

Allowing the script to continue brings a risk of collision. Ensure that the next steps in the scripts are performed away from the Loading ID 5-grid grids.

- Always include a **Wait for Finish** command before the steps in the script requiring arm access to the Loading ID 5-grid grids (e.g., pipetting from the bar-coded tubes or transporting a tube out of a Loading ID 5-grid tube runner).

Generate ID:

Fig. 108: Generate ID

In case of an unread barcode, with the option **Allow to continue with errors** selected, a unique ID is automatically generated. A default will be suggested with prefix and sequential numbering with <IDX5> that adds a 5-digit sequential number (e.g., 00001). The prefix and sequential number can be changed. If only a text is defined, it will add a single sequential number to that (e.g., barcode-1). The prefix and sequential numbering can be customized.

Preview: Click on the preview button, then load a tube runner on a Loading ID grid to see the impact of the chosen settings on the barcode reading results on the touch monitor and on the LED color.

Color: Define a custom color for the grid LEDs signaling the invitation to load. Default is set to blue. Preview will show the configured LED color for 5 seconds.

CSV-Output: Define a path and file name for the automatic generation of a CSV-File containing all read tube barcodes. In the current vControl version, the headers of the CSV output file do not always correspond to the data in the corresponding column. The file contains the following data for each position in the runner in comma separated columns:

GridPos	Grid position of the runner
SiteOnGrid	Always '1' for tube runners
TubeNumber	Tube position in the runner
ConfigName	Barcode of the runner (the string from Custom Attribute Loading ID 5-grid_tUbeRunnerID)
RackLabel	Always 'N/A', currently not used
CarrierBarcode	Barcode of the tube, '\$\$\$' if no tube was detected or '***' if the barcode could not be read
SRCRackID	Always 'N/A', currently not used

11.5.4 Allow Removal

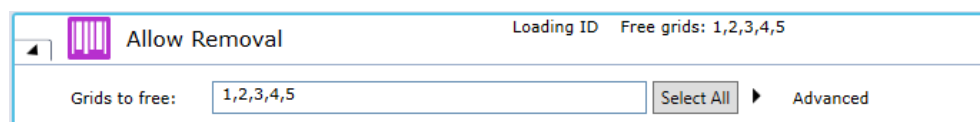


Fig. 109: Allow removal command

Allow operator to unload runners and tubes during the course of a run. The blinking blue (or customized) LED signals the runners that are ready to be unloaded. grids are allowed to unload. The monitoring for these grids will be deactivated. There is no dialog displayed on the touch monitor.

The tubes will be flagged as not present in the worktable. This allows the LabwareCount function to work correctly. Take care not to use Allow Removal before all needed pipetting actions have been executed.



*After a **Allow Removal** command, insert a **Wait for Finish** before the insertion of a second **Start Loading** command in the vControl script.*

Grids to free: Define the grid positions where unloading is allowed. Enter ranges or single grid positions into the textbox. Select all, if all available grids shall be unloaded. Enter a single grid, individual grids separated by commas, or a range of grids separated by a hyphen (e.g., 1,2,4-6).

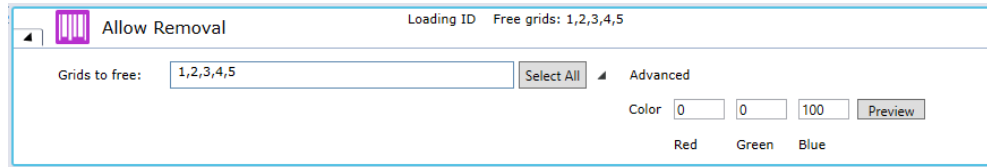


Fig. 110: Grids to free

Advanced

Define a custom color for the grid LEDs to be used for removing is allowed.

NOTICE

Avoid unexpected Active Stops

To avoid unexpected Active Stops of the system, place an **Allow Removal** command as soon as all required steps (e.g., liquid transfer) are successfully executed with the selected runners and tubes.

11.5.5 Wait for Finish

Wait for Finish: The **Wait for Finish** command if the option **Continue** while **Loading is selected**. It is used after a **Start Loading** command to pause the vControl Script until the loading of the runners is completed.

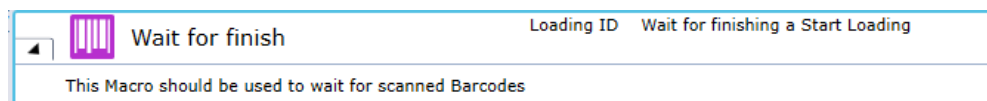


Fig. 111: Wait for finish command

11.6 DeckCheck

The DeckCheck command provides the user with the functionality to execute a discrepancy check on the worktable during runtime. The DeckCheck command can be used multiple times in the script, where each execution is independent to the other executions, i.e., each script command has its own set of pictures which will be used by the DeckCheck function during execution. Discrepancies are not logged as errors. Therefore, the result of the discrepancy check will not have any impact on the result of the run.

11.6.1 DeckCheck Network Settings

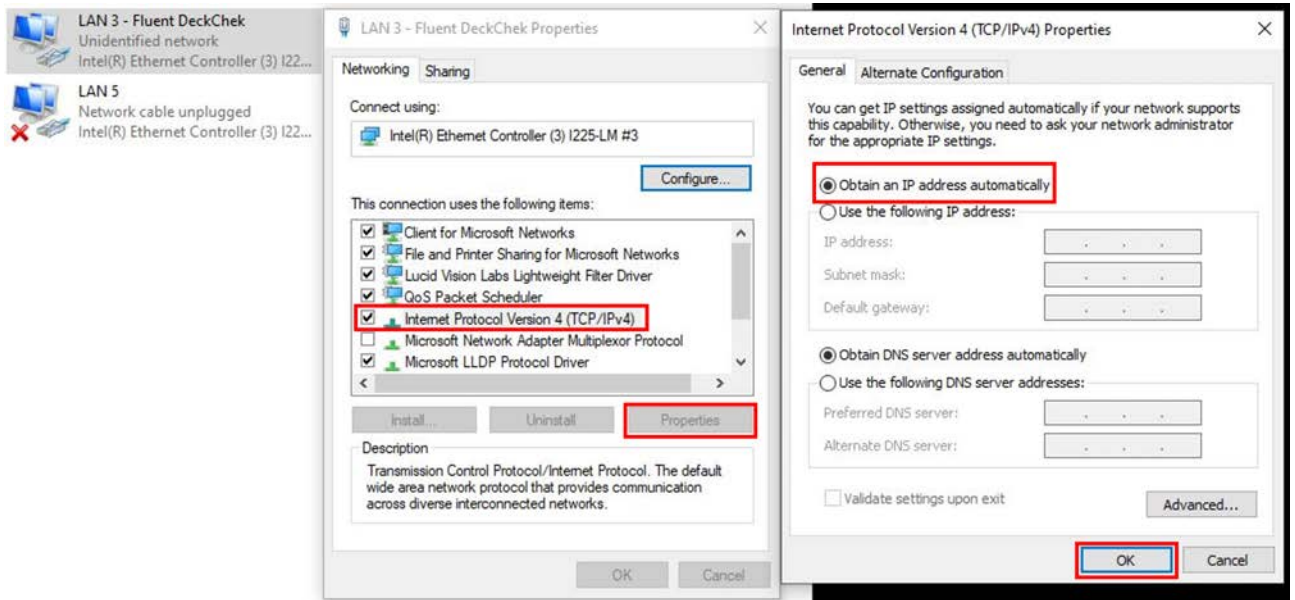


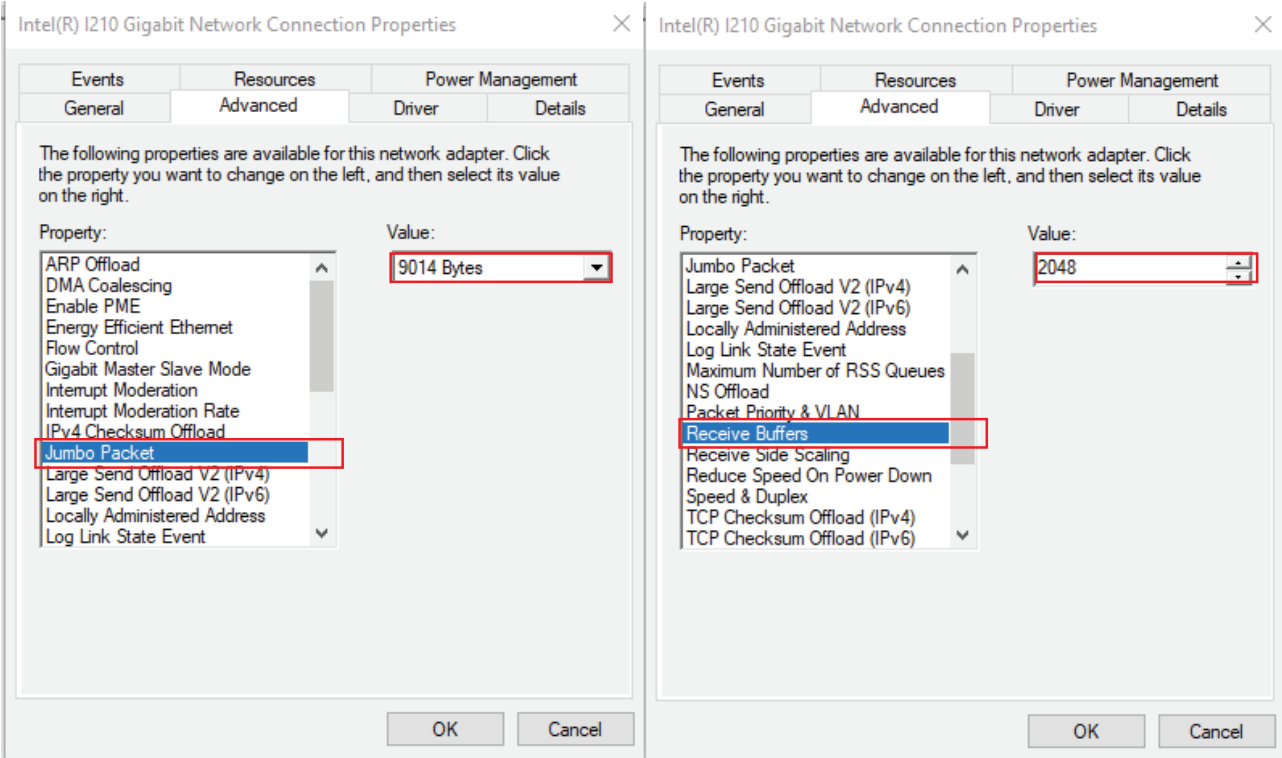
Fig. 112: DeckCheck Network Settings

The following settings are to be performed to use DeckCheck within vControl:

1. Open the properties of the respective LAN connection.
2. In the **Networking** tab select **Internet Protocol Version 4** and click **Properties**.
3. In the next window select **Obtain an IP address automatically** and click **OK**.
4. Confirm the changes in the LAN properties window by clicking **OK**.

11.6.2 Ethernet Adapter Settings

Then using DeckCheck, the following settings for the Ethernet adapter are recommended.



**Select Jumbo
Packet**

Enabling Jumbo Frames on the Ethernet adapter allows a packet size of up to 9014 bytes to be set on the camera. The larger packet size will enable optimal performance on high-bandwidth cameras, and it usually reduces CPU load on the host system. Please note that to set a 9014-byte packet size on the camera, the Ethernet adapter must support a Jumbo Frame size of 9014 bytes or higher.

**Adjust Receive
Buffer Size**

A receive buffer is the size of system memory that can be used by the Ethernet adapter to receive packets. Increasing the receive buffer size will increase the performance but also result in increased system memory usage.

11.6.3 Edit Time



Parameters

Title: The text entered here is displayed on the touch monitor at run time and can be customized to provide information to the operator.

Show always: If this option is selected, the DeckCheck screen is shown on the touch monitor during runtime whether any discrepancies were detected or not. If this option is not selected, the DeckCheck screen will only be shown in case discrepancies were found.

Take Reference Scene: A reference scene is a set of images (one for each camera) that serves as reference for a later comparison with live images. After clicking on **Take reference scene** each camera takes a picture of the worktable. The instrument may execute arm movements to move out of camera view.

Region(s) of Interest: A region of interest defines a section of the worktable where discrepancies are of interest. Discrepancies outside any regions of interest will be ignored. Each camera has its own perspective, and its own set of regions of interests. Select each camera on the left side of the main image to define regions of interest for that view. A region of interest can be defined by dragging a rectangle on the reference image with the mouse. This rectangle can be modified in size and can be deleted by pressing on the **X** on the region(s) of interest list on the right. Each region of interest can be given a name in and a confidence threshold.

Allow Operator to ignore and continue: If this option is selected, the Operator can press **Continue** at runtime without resolving detected discrepancies. If this option is not selected, the run can only be continued if all discrepancies have been corrected.

Confidence threshold (%): The discrepancy detection is based on neural networks, that predict discrepancies between the reference image and the live image with certain confidence. A low confidence threshold allows more potential discrepancies to be detected, however can also lead to more false positive detections meaning it is more likely to detect noise that is no real error. A high confidence threshold is the opposite, it will only show up discrepancies that have high certainty of being real errors, meaning it might miss some potential discrepancies. The values in % for each of the three levels can be set in the Settings > Configure System > DeckCheck.

Note:

- If no region of interest is defined for any camera, the entire image for each camera is considered as region of interest with a medium confidence threshold value.
- If at least one camera has a region of interest defined, all other cameras without any region of interest will be disregarded.
- For best discrepancy detection it is recommended to define the same region of interest for each camera.
- It is recommended to only cover critical areas of the worktable with a region of interest. Areas outside of the worktable like the side panels and the cabinet should not be covered to avoid detection of unwanted discrepancies.

There is no limit to the number of regions of interest that can be set. The number of regions set in one command will not affect the speed of operation.

11.6.4 Configuration

The DeckCheck settings section can be found under: **Configure System, General Settings** DeckCheck tab. A user with the corresponding user rights, can configure the confidence threshold settings of each of the three levels low, medium and high. The default recommended threshold values based on empirical data are 80%, 90%

and 95% for low, medium and high confidence respectively. For lower thresholds, reliability of a correct detection result decreases drastically and thresholds below 50% are considered to not add much value anymore to a correct detection.

Disable DeckCheck: If this option is selected, all DeckCheck commands in the script will be ignored and vControl just jumps to the next line without executing the command.

In **Configure System, Devices, Home positions**, the positions the arms move to in order not to block the cameras' view during image acquisition may be defined

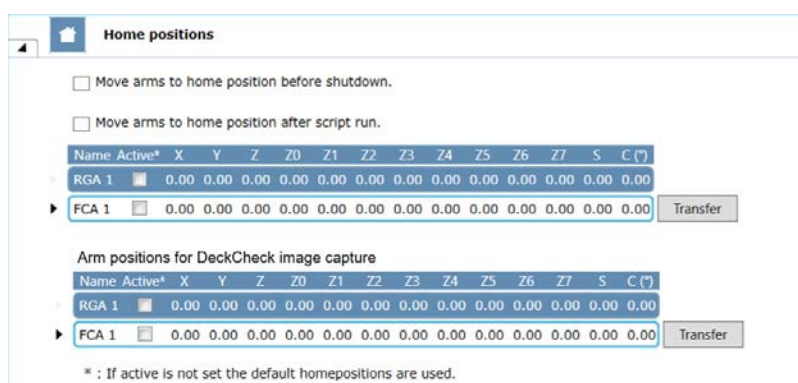


Fig. 113: Home positions

11.6.5 Run Time

During DeckCheck command execution the system performs the following steps before displaying the images:

- Arm movement (if necessary, and if the command is not executed in the scope of a script loading guidance) to allow unobstructed camera views on the worktable.
- Image acquisition
- Evaluation and validation of the images
- Displaying of the images

On the touchscreen the following elements are then displayed:

- Reference image
- Live image

The live and reference images are alternating and detected potential discrepancies are marked with a red square. If there are three cameras on the system, the first camera from the left with any discrepancies will be displayed. If no discrepancies are found, the left camera will be displayed.



Parameters

The speed of the DeckCheck execution until image display depends on the PC hardware configuration

Camera View: To navigate between the cameras there is a button for each camera. Clicking on the respective button (left, center, right) will display the corresponding live and reference image of that camera.

Pause Alternate or Resume Alternate: Starts and stops the alternation between the reference and live images.

Check: Performs a re-validation of the live image. The DeckCheck command is executed again and runs through arm movement (if necessary), image acquisition, validation of pictures against reference pictures and display. This button can be used if the user needs feedback on whether all the discrepancies are resolved. An automatic re-validation is executed when closing the front door.

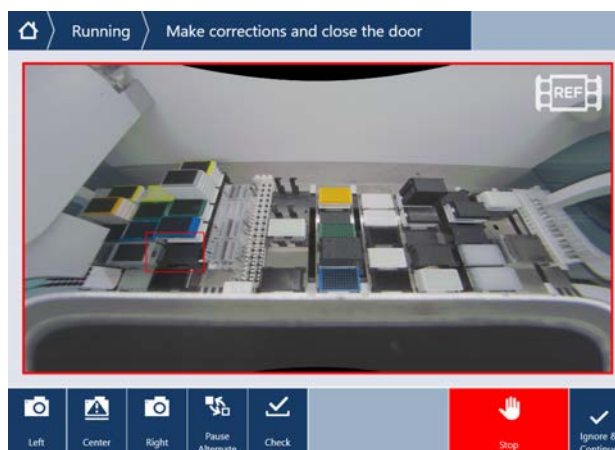
Stop: The run is stopped. The user has the possibility to retry the run or abort it.

Ignore & Continue: This option is only available in case any discrepancies are found and **Allow operator to ignore and continue** has been selected in the DeckCheck command. Pressing this button will log the decision and any discrepancies will be ignored.

Continue: If no discrepancies are found, this button appears to continue the run.

Layouts

Discrepancies have been found:



The area of error depicts the region(s) where discrepancies were found. The area of errors will be highlighted on the live and the reference images with a red square.

The cameras with any detected discrepancy are highlighted with an exclamation mark on the respective camera buttons. Furthermore, a red border indicates that this camera has encountered one or more discrepancies.

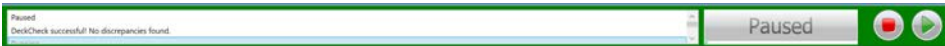
No discrepancies have been found:







A green border indicates that no camera has encountered any discrepancy.

Notifications

Notification in Runtime Controller, if no discrepancies are found.



Notification in System Trace View if discrepancies are found.

System Trace View						
Status	Start Time	End Time	Module	Macro	Object	Message
	17:31:29	17:31:29	SystemSW-7428			Run paused.
	17:31:28	17:31:28	SystemSW-7428			DeckCheck discrepancy detected
	17:31:03	17:31:03	SystemSW-7428	Initialization		Instrument initialization finished.
	17:31:03	17:31:03	SystemSW-7428	Initialization		Driver initialization finished.

LastDeckCheckResult Function

The function „LastDeckCheckResult()“ can be used to retrieve the result of the last DeckCheck script command. The return value is of type string.

Tab. 4: DeckCheck Result

DeckCheck Result	Type	Description
“true”	string	DeckCheck did not find discrepancies
“false”	string	DeckCheck found discrepancies

11.6.6 Get notified with Introspect

Introspect sends the user a notification if any discrepancies have been detected. For more information about the use of Introspect, please refer to <https://www.tecan.com/knowledge-portal/digital-solutions>.

11.6.7 Capture Worktable



The 'Capture Worktable' command of the Deck Check driver captures pictures from all cameras on the instrument. The system performs an arm movement (if necessary) to allow unobstructed camera views on the worktable. The captured worktable images are saved under 'C:\ProgramData\Tecan\LoggingServer\LogFiles\DeckCheck Pictures' and can be collected with the Snapshot Tool.

11.7 Magellan

11.7.1 Measure



The measure command of the Magellan driver can be used to execute a measurement with a Tecan Reader controlled by Magellan.

Parameters

MP: This field contains the name of the micro plate, that is to be measured by the Tecan reader. Either type your name or click in the Worktable editor window (Worktable Click).

Edit: Opens an editor to select the Magellan method to be executed for the measurement.

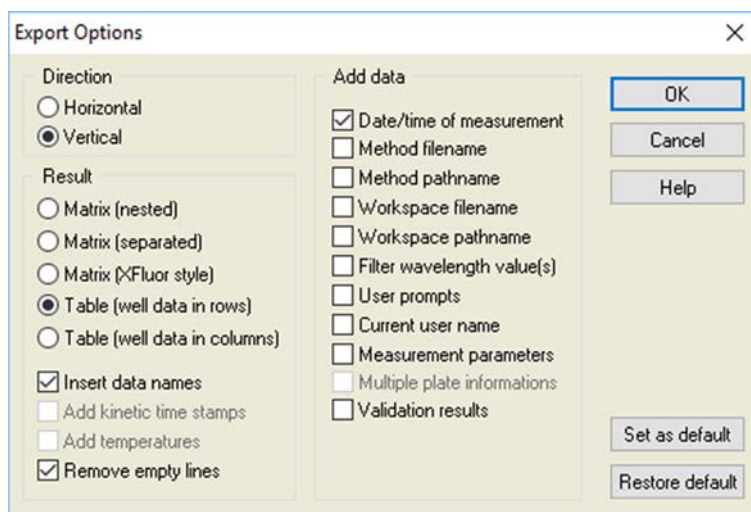
The following are effective whenever the Measure command is executed:

- A previously saved standard curve defined in the layout of the executing Magellan method is saved to a standard curve file and its graphical visualization to a bitmap file. Magellan uses the following target folder:
C:\Users\Public\Documents\Tecan\Magellan Pro\std
- A previously saved standard curve file, that was previously saved, to apply gain and z-Position of the standard curve to the executing Magellan method. Magellan uses the following source folder:
C:\Users\Public\Documents\Tecan\Magellan Pro\std
- English is used as the standard language for the measurement output file (.asc)
- If the 'Use standard output format' checkbox is unchecked, the output file format is as defined in the Magellan method
- If the 'Use standard output format' checkbox is checked, the following export settings are used as standard format for the measurement output file (.asc):

Export Options:

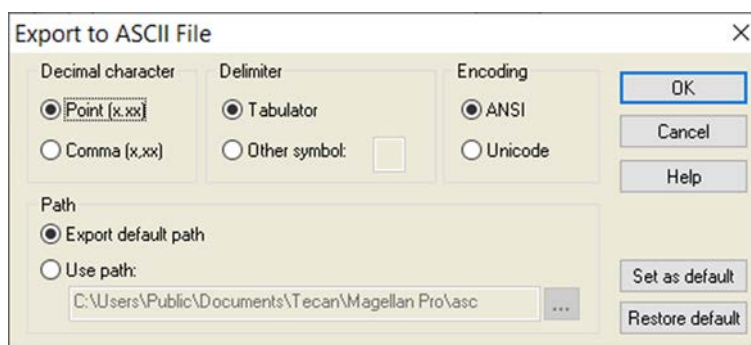
- Direction: Vertical

- Result: Matrix (separated)
- Add data: Date/Time of measurement
- Insert data names
- Remove empty lines



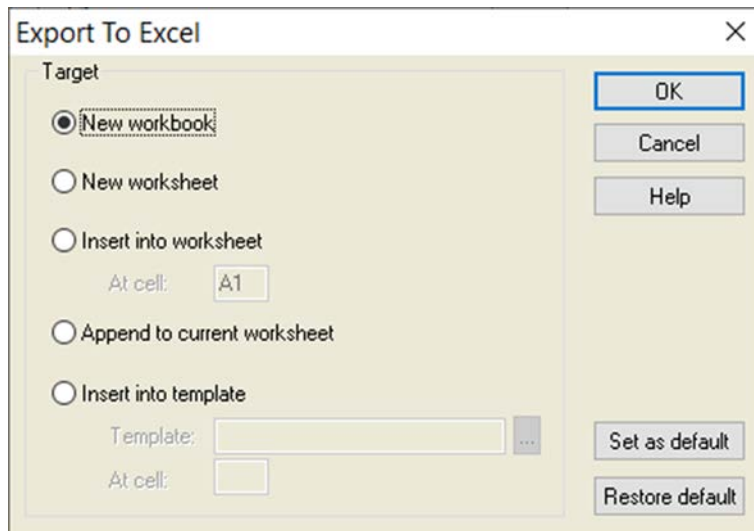
Export to ASCII file

- Decimal character: Point (x.xx)
- Delimiter: Tabulator
- Encoding: ANSI
- Path: Export default path



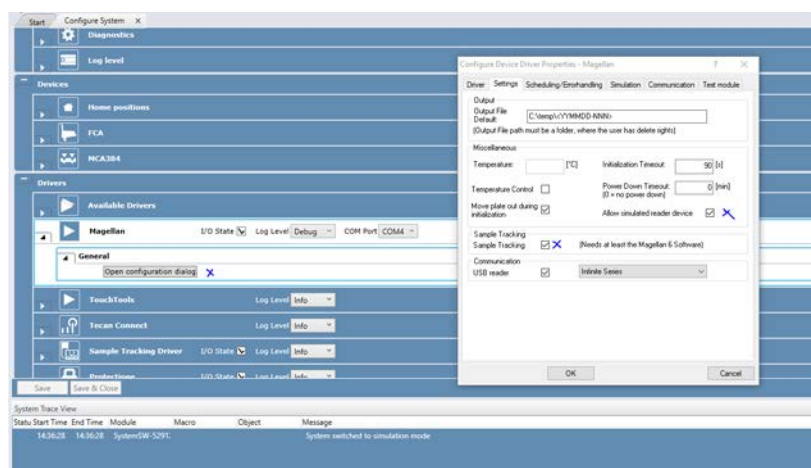
Export to Excel (requires that Excel is installed on the PC)

- New workbook



11.7.2 Driver Settings

Under Configure System > Magellan driver settings ('open configuration dialog') it's important that Sample Tracking is enabled. If not, this is the main reason why Sample IDs will not be available for Magellan and later on Store Well Command. In addition, for running the Magellan driver in simulation the checkbox 'allow simulated Reader device' must be activated. Restart vControl after these changes.



11.8 Inheco Devices

11.8.1 MTC Controlled Devices

The following inheco devices are controlled by STC/MTC:

- CPAC
- CPAC2TEC
- Teleshake 95 AC

- Thermoshake AC

Plate incubator device

- Single Plate Incubator w/o shaking capability

11.8.1.1 HEATPAC / CPAC / CPAC2TEC script commands

Enable Temperature Control

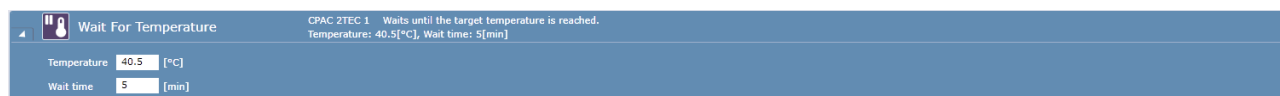


This command sets the specified target temperature on the device and enables the heating/cooling.

Temperature ranges:

- HEATPAC: ambient to +135°C
- CPAC: +4°C to +70°C
- CPAC2TEC: +4°C to +110°C

Wait For Temperature



This command can be used to ensure that the device will be on the requested temperature in the given time.

If the temperature cannot be reached in the specified time, an error dialog appears with options to ignore, retry, or cancel.

Disable Temperature Control



This command stops the temperature regulation on the device.

Teleshake 95 AC / Thermoshake AC script commands

Enable Temperature Control




This command sets the specified target temperature on the device and enables the heating/cooling.

Temperature ranges:

- Teleshake 95 AC: ambient to +125 °C
- Thermoshake AC: +4°C to +70°C

Wait For Temperature

 Wait For Temperature	Teleshake 95 AC: Waits until the target temperature is reached. Temperature: 60.8[°C], Wait time: 3[min]
Temperature: <input type="text" value="60.8"/> [°C]	
Wait time: <input type="text" value="3"/> [min]	

This command can be used to ensure that the device will be on the requested temperature in the given time.

If the temperature cannot be reached in the specified time, an error dialog appears with options to ignore, retry, or cancel.

Disable Temperature Control

 Disable Temperature Control	Teleshake 95 AC: Disables temperature control.
--	--

This command stops the temperature regulation on the device.

Start Shaking

 Start Shaking	Teleshake 95 AC: Starts shaking with a defined shaking frequency. Revolutions: 300[rpm]
Revolutions: <input type="text" value="300"/> [rpm]	

This command sets the revolution of the shaker and starts the shaking.

Revolution range: 150 to 3000 [rpm]

Stop Shaking

 Stop Shaking	Teleshake 95 AC: Stops shaking.
---	---------------------------------

This command stops the shaking on the device.

11.8.1.2 Plate Incubator Script Commands


Enable Temperature Control

 Enable Temperature Control	Inheco Incubator Shaker: Enables temperature control. Temperature: 57[°C]
Temperature: <input type="text" value="57"/> [°C]	

This command sets the specified target temperature on the device and enables the heating.

Temperature range: from ambient temperature +5K up to +80°C

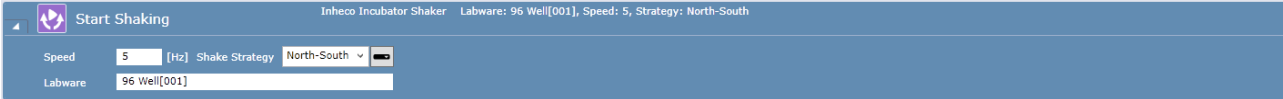
Wait For Temperature

 Wait For Temperature	Inheco Incubator Shaker: Waits until the target temperature is reached. Temperature: 60[°C], Wait time: 5[min]
Temperature: <input type="text" value="60"/> [°C]	
Wait time: <input type="text" value="5"/> [min]	

This command can be used to ensure that the device will be on the requested temperature in the given time.

If the temperature cannot be reached in the specified time, an error dialog appears with options to ignore, retry, or cancel.

Start Shaking

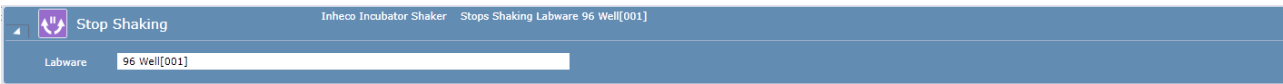


This command sets the speed and shake strategy of the incubator starts the shaking on the device where the given labware is placed.

Speed range: 6.6 to 30 [Hz]

Shake strategies: North-South, East-West, Round

Stop Shaking

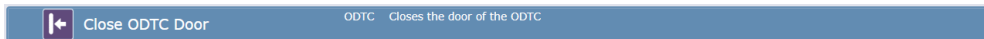


This command stops the shaking on the device where the given labware is placed.

11.8.2 Inheco ODTc

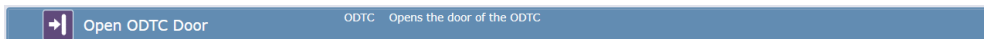
The following commands are loaded when an Inheco ODTc is connected and configured in vControl.

Close ODTc Door



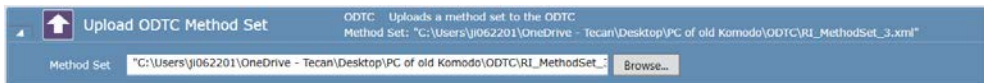
The Close ODTc Door command will close the door of the ODTc.

Open ODTc Door



The Open ODTc Door command will open the door of the ODTc.

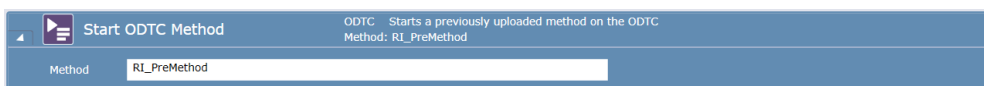
Upload ODTc Method Set



The Upload ODTc Method Set command will upload a method set to the ODTc Driver which was created outside of vControl. The method set can contain a list of methods which can be executed by the ODTc.

Method Set: The file path of the method set file. The method set must be in XML format. The "Browse..." button will open a file browser to select a method set file.

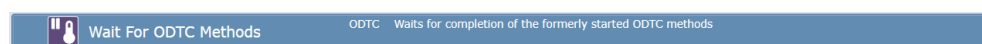
Start ODTc Method



The Start ODTc Method will asynchronously start a Method which was previously uploaded to the ODTc Driver using the Upload ODTc Method Set command. If the method name was not uploaded using the Upload ODTc Method Set command, the method will not run.

Method: The name of the method which was uploaded in one of the method set files.

Wait for ODTc Methods



The Wait for ODTc Methods will wait for all previously started ODTc Methods to finish their execution. Once all methods are complete, the script execution will continue.

11.9 Programming

11.9.1 If

With this programming command, you can branch the execution of your script depending on the **Condition** parameter.

If the specified Boolean condition is met (numerical result not equal to 0), the statements within the **If** group are executed. Otherwise, the execution continues in the following branch—either in the optional **Else** group, or after the end of the **If** group.

11.9.2 Comment

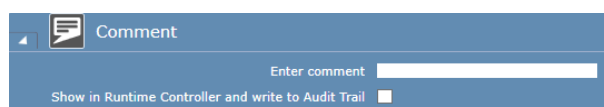


Fig. 114: Programming: Comment

This command is used to insert a comment for a script command or for some other reason. It is used to annotate the script and is not executable or visible to the operator. Use the “User Prompt” [248] command instead if you want to prompt the instrument operator. When you run the script, the comments are shown in the Runtime Controller of vControl. If you choose suitable comments, this can help the operator to monitor the progress of the script as it runs.

Settings

Comment: Enter a comment. If the comment will be used for the Audit Trail, check the check box.

11.9.3 Group

The Group command is used to group together script lines to make long scripts easier to read.

Description

Specify a descriptive name of your choice for the group.

11.9.4 Loop

With the Loop command, you group commands and set Begin Loop and End Loop automatically.

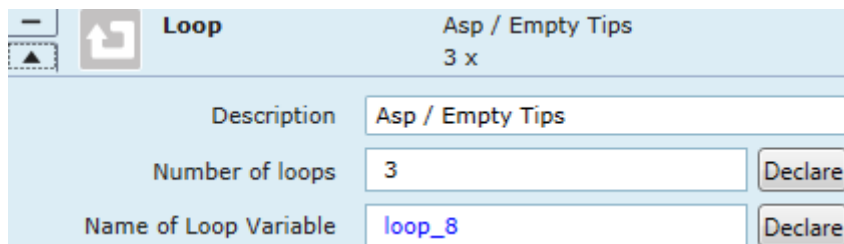


Fig. 115: Programming: Loop

Description

Name of Loop Variable: You can give the Loop a meaningful name or short description.

Number of loops: Specify the required number of times the loop should be repeated (range: 1–999).

11.9.5 Leave Loop / Group

With the Leave Loop command, you can leave the current loop. It skips all possible remaining cycles and continues with the statement behind it.

This programming command has no parameters.

11.9.6 Next Cycle

With the Next Cycle command, you can control the workflow of the script. Next Cycle leaves the current Loop cycle and jumps to the first command in the Loop. On leaving the last Loop cycle, it jumps to the first command after the Loop instead.

This programming command has no parameters.



If the command is outside of a loop in runtime the command will have no effect.

11.9.7 Else

The **Else** branch of a corresponding **If** statement.

If the condition specified in the complementary **If** statement is not met (numerical result equal to 0), the **Else** group of this **If** statement gets executed. Only one **Else** group can be added after an **If** group. Check that all **Else** branches are preceded by an **If** statement and that there is only one **Else** branch per **If** statement.

11.9.8 User Prompt

Script command to issue a message with requested user feedback.

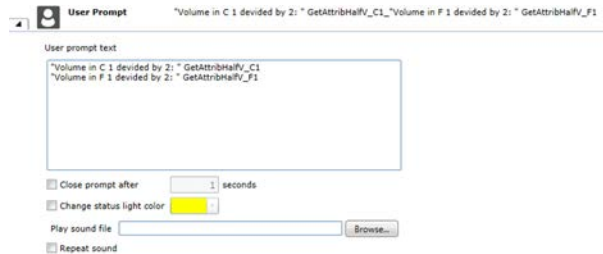


Fig. 116: Programming: User Prompt



Fig. 117: Programming: User Prompt example

11.9.9 User Input

Use this command to request input from the operator for a variable value.

Name

Declare: Each variable which is used in a User Input command has to be declared first. Type the Name of your variable, and then click **Declare**.

Query Prompt: Type a question to the user.

Limit operator to allowed range: Specify maximum and minimum values for the numeric variable.

11.9.10 Set Variable

Use this command to set the value of a variable.

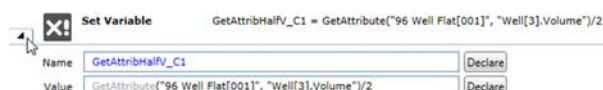


Fig. 118: Programming: GetAttribute Set Variable

The Declare buttons provide a convenient way to declare new variables. Please check the correctness of the declaration in the Variables section of the Script Editor afterwards.

You can use other variables or expressions in the Value field to calculate the new value for the variable. See [“Functions and Operators” \[306\]](#) Functions and

Operators for a list of the available operators and functions.
Note that some functions can be used to manipulate the system database settings. For example, the SetAttribute function can be used to set the barcode information for a labware.



Always use double quotes ("") when assigning values to string variables.

For example, when a String variable is used to hold a barcode, the assigned value must use double quotes like "123456" because the barcode is a String, not a number.



Please note that if the non printing characters (ASCII characters from 0 to 31 except 9, 10 and 13) are used in the input fields there will be no context check error shown. The non printing characters will be ignored and they will not be saved.

11.9.11 Import Variable

The **Import Variable** command is used to import values for script variables from a file during script runtime. You can import values for variables which are already declared (defined) in the script. The import file can be either a comma-separated text file (TXT or CSV) or an Excel file (XLS). To read XLS files, Excel must be installed on the PC.

11.9.12 Export Variable

The **Export Variable** command is used to export the values of script variables to a file during script runtime. The export file can be either a comma-separated text file (TXT or CSV) or an Excel file (XLS).

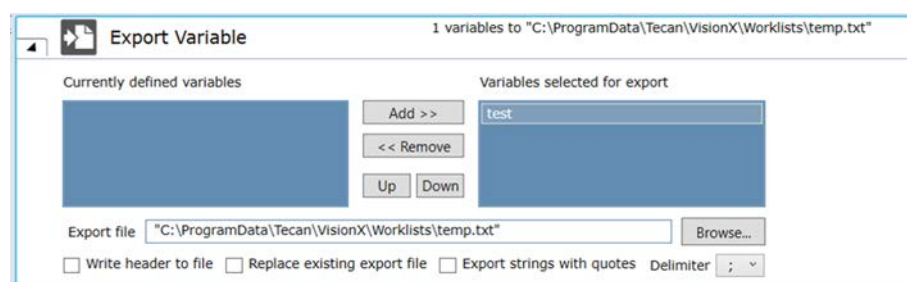


Fig. 119: Export Variable

The **Export strings with quotes** checkbox allows to choose whether to export variables of type string with enclosing quotation marks or without them.



Special characters such as °, μ as well as any superscript numbers will be converted to '?' characters when exported to TXT or CSV file formats.

11.9.13 Start Timer

The **Start Timer** command re-starts one of the 100 internal timers in vControl. If required, you can use this command to re-start a specified timer several times in the same script.



Fig. 120: Programming: Start Timer

Timer number

Specify the number of the timer you want to use or specify a numeric variable (range: 1 - 100).

11.9.14 Wait for Timer

Timer number: Specify the number of the timer you want to use or specify a numeric variable (range: 1–100).

Duration [sec]: Specify the time duration which has to expire since the selected timer was started or reset or specify a numeric variable (range: 0.02–86,400 s). The script will be paused until the specified time duration has been reached.

On active stop of the run the timer will not be paused or restarted. It will continue counting down.

11.9.15 Wait

Use this command to have the instrument **Wait** for a specific time duration.

Duration [sec]: Specify the time duration which has to expire since the selected timer was started or specify a numeric variable. With the Declare button you can declare a variable using the current parameter of "Duration".

Display Countdown

Indicates whether to display the countdown when executing the command. In process scripts, this option is deselected by default. On retry after an active stop, the command behaves differently depending on the used option. Selected: Countdown is restarted. Deselected: Wait command is skipped.

11.9.16 Subroutine

The **Subroutine** command executes another script as part of the current script.

Browse: Click this button to look for existing vControl scripts or modules and to choose the one you want to use as a subroutine.

Start Subroutine and wait for it to finish: This option causes the main vControl script to wait for the subroutine to finish before continuing.

Start Subroutine and continue main script: With this option, the main script continues without waiting for the subroutine to finish (asynchronous subroutine).

Wait for previously started Subroutine to finish: This option causes the main vControl script to wait for the specified, previously started subroutine to finish before continuing.

Transfer variable values to Subroutine: this list contains the default values of variables with scope Parameter in the Subroutine. The values can be modified and will then be used by the Subroutine. If the value of a variable from the main script shall be passed to the Subroutine, enter the variable name in this field instead.

Values for variables of type String are wrapped in double quotes. Empty string values will lead to context check errors.

This option is not available if "Wait for previously started Subroutine to finish" (asynchronous subroutine) is selected.

Only variables with scope Parameter are shown in the "Transfer values to subroutine" section. It is not possible to remove variables with scope Parameter, however, additional variables may be added to the list to pass them to the Subroutine.

- **Value from main script:** this list contains the values of variables with scope Parameter in the Subroutine. If the value of a variable from the main script shall be passed to the Subroutine, enter the variable name in this field instead
- **Variable in subroutine:** this list contains all variables with scope Parameter used in the main script as well as the Subroutine

Transfer variable values from subroutine to main script: variables whose values shall be transferred from the Subroutine to the main script after the Subroutine has been executed can be manually entered here. This option is not available if "Start Subroutine and continue with main script" (asynchronous subroutine) is selected.

11.9.17 Start Application

Wait for termination and store return code in a variable.

11.9.18 Start VB

Use this command to execute a Visual Basic (VB) script from within a script.

vControl supports VB .NET instead of Visual Basic Scripting (VBScript) like EVOware. VB .NET offers many more options for interactions, but may be more complex. To get an idea of how to use this, take a look at the examples.

GetAndSetNumVars.vb

```
Imports Microsoft.VisualBasic
Imports System
Imports Tecan.Core.Scripting
Public Class TestClass
Implements IScriptObject
Private Host As IScriptingHost
Public Property ScriptingHost() As IScriptingHost Implements
IScriptObject.ScriptingHost
Get
Return Host
End Get
Set(ByVal value As IScriptingHost)
Host = value
End Set
End Property
Public Sub Execute() Implements IScriptObject.Execute
Dim factor1 As Double
Dim factor2 As Double
Dim result As Double
factor1 = Host.GetVariable("numValue1")
factor2 = Host.GetVariable("numValue2")
result = factor1 * factor2
Host.SetVariable("result", result)
```



```
End Sub
End Class
```

GetAndSetForce.vb

```
Imports Microsoft.VisualBasic
Imports System
Imports Tecan.Core.Scripting
Public Class TestClass
Implements IScriptObject
Private Host As IScriptingHost
Public Property ScriptingHost() As IScriptingHost Implements
IScriptObject.ScriptingHost
Get
Return Host
End Get
Set(ByVal value As IScriptingHost)
Host = value
End Set
End Property
Public Sub Execute() Implements IScriptObject.Execute
Dim force As Integer
force = Host.ResolveExpression("GetAttribute(""96 Well
Flat[001]", ""Force"")")
force += 10
Dim expression As String = String.Format("SetAttribute(""96
Well Flat[001]", ""Force"", {0})", force)
Host.ResolveExpression(expression)
End Sub
End Class
```

The VB.NET program has to implement a certain interface in order to be loaded and compiled by vControl. This is a small sample program:

```
Imports Microsoft.VisualBasic
Imports System
Imports Tecan.Core.Scripting

Public Class TestClass
Implements IScriptObject

Private Host As IScriptingHost

Public Property ScriptingHost() As IScriptingHost Implements
IScriptObject.ScriptingHost
Get
Return Host
End Get
Set(ByVal value As IScriptingHost)
Host = value
End Set
End Property

Public Sub Execute() Implements IScriptObject.Execute
// INSERT YOUR CODE HERE
```

```
End Sub
```

```
End Class
```

11.9.19 End Script

The **End Script** command is used to stop the script prematurely and to specify a return code (error code) for the script. It is typically used together with error recovery commands.

11.9.20 On Error Goto

The **On Error Goto** command is used to jump to a specified line in the script (error handler) if an error occurs during script runtime. Error recovery commands then allow the script to react automatically to the error.

11.9.21 Error Handler

This special group statement represents an error handling routine that can be used by the **On Error Goto** statement.

11.9.22 Raise Error

With this programming command you can raise an error.

Parameters

Severity Level: Drop-down list box to define the severity level of the to be raised error. Range: ["Recoverable", "NonRecoverable"]. **Default value:** "Recoverable".

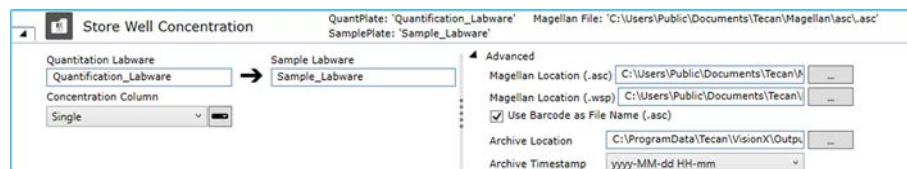
Error Message: Type the error message to be displayed.

11.9.23 Store Well Concentration

The Store Well Concentration Command is used to save the concentration data from a Magellan ASC File to a selected sample labware.

The quantification labware is used to calculate the dilution factor used during the script execution and apply this to the concentration values of the sample labware accordingly.

The concentration data from the sample labware will then also be sent to Sample Tracking during the execution of the command.



Parameters

Quantitation Labware: This field specifies the quantification labware used during the execution.

Sample Labware: This field specifies the labware containing the samples.

Magellan Location (.asc) (.wsp): These fields specify the path to the Magellan files used to execute the command.

The Checkbox “**Use Barcode as File Name**” can be checked, if the file name is the same as the barcode of the labware used.

Archive Location: This field specifies the file path of where the Magellan files will be archived after the command is executed.

Archive Timestamp: This drop-down list will change the format of the timestamp added to the archived Magellan files.

11.9.24 Confirm Standard Curve

The Confirm Standard Curve is used to confirm that a displayed standard curve matches the user expectations.

Edittime

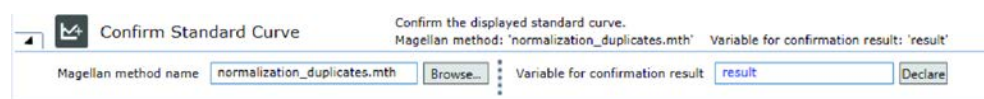


Fig. 121: Confirm Standard Curve

Parameters:

Magellan method name: The name of the Magellan method that has been used for the measurement of the standard curve. The user can select the Magellan method using the “Browse” button. This opens the default location for Magellan methods in the file system. The path to Magellan method in the file system is automatically added by the system.

Variable for confirmation result: An integer variable that is used to store the result of the confirmation dialog.

Runtime

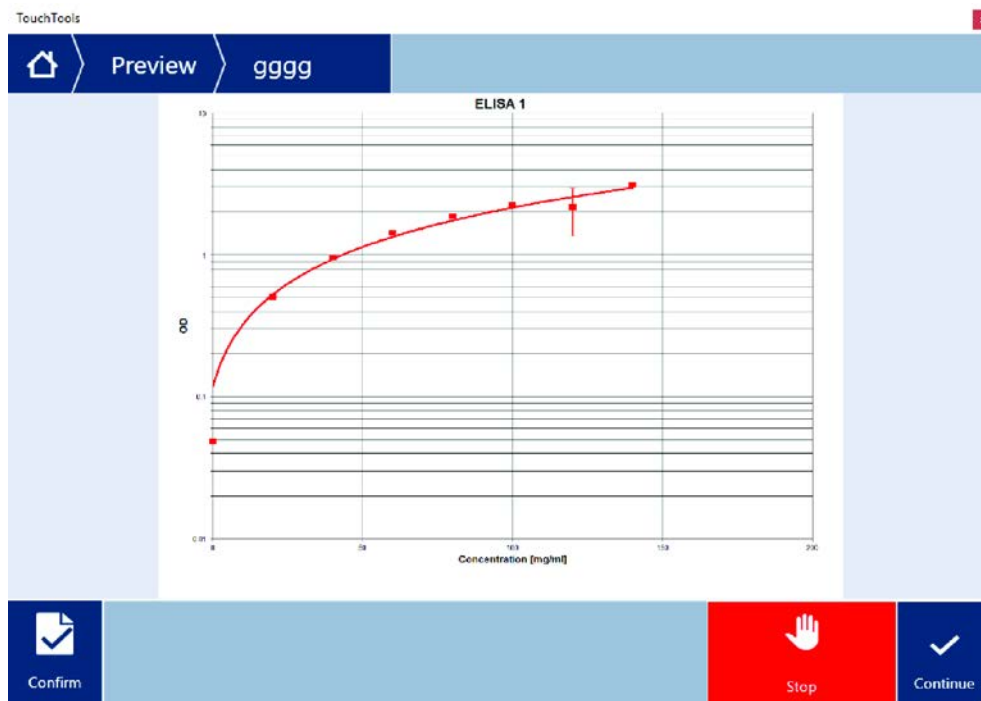


Fig. 122: Runtime

Buttons:

Confirm: Confirms that the displayed standard curve matches the expectations.

Stop: Stops the current run.

Continue: Continues the run.

Result

If the run is continued with confirmation, the value of the confirmation result variable will be set to 1.

If the run is continued without confirmation, the value of the confirmation result variable will be set to 0.

11.9.25 On System Prompt

The **On System Prompt** command allows to automatically respond to an error dialog if an error occurs during script runtime. The settings made in this command are valid until the current script ends or until they are overridden by a subsequent **On System Prompt** command.

The **On System Prompt** command does not allow handling any of the following dialogs:

- User Prompt
- User Input
- RUP

If a runtime error is encountered that matches the filter condition, the system automatically presses the specified button. The dialog is suppressed entirely—i.e., it is neither visible in the application nor on the touch screen.



Fig. 123: On System Prompt

Filter Condition

The **Filter Condition** parameter specifies the conditions that must be satisfied for handling an error (e.g., filtered by their error ID). If this parameter is left empty, the specified handling applies to all runtime errors.



*It is possible to define multiple error handlings by using several **On System Prompt** commands with different **Filter Condition** parameters. If the **Filter Condition** parameters overlap, the most recent one in the script applies.*

Error Counter

The **Error Counter** parameter allows specifying a variable that is used to count the number of errors being handled. If this parameter is left empty, no counting will be performed.

Button to press

The **Button to press** parameter is required to be filled in. The button to be pressed can be specified either by its button text (case insensitive) or by its index—starting with 1.

11.9.26 Switch UV Light On

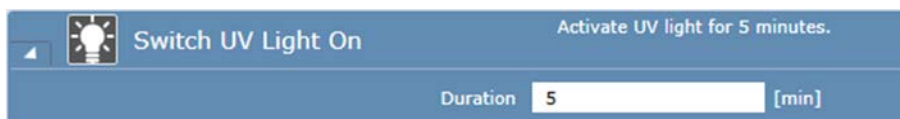


Fig. 124: Switch UV Light On

This command is only available if the UVC light option, which is used for decontamination, is installed.

To avoid shadow casting—and, hence, impaired decontamination—the instrument arms should be moved appropriately during the UVC radiation cycle by the according commands (e.g., Move Arm (FCA), [“Move Arm \(RGA\)” \[224\]](#), [“Subroutine” \[251\]](#)). Labware or other objects that might impair the decontamination have to be removed from the worktable. Furthermore, UVC light can damage plastic and reagents.

Parameters

Duration: Define the irradiation duration in minutes (range: 1–30 minutes).



*Each On/Off switch cycle reduces lifetime of the UVC light by 30–60 minutes. The command **Switch UV Light On** cannot be repeated within a 5-second interval. If this is attempted anyway, an error message is shown.*

When opening the front safety panel or the diluter panel, the UVC light is switched off automatically and an error message is shown.

The UVC light is switched off automatically after the defined irradiation duration—even if vControl has stopped running or the connection to the instrument failed.

11.9.27 Switch UV Light Off



Fig. 125: Switch UV Light Off

This command is only available if the UVC light option, which is used for decontamination, is installed.

This command switches the UVC light off.



*The use of the **Switch UV Light Off** command is optional. The UVC light is switched off automatically after the defined irradiation duration—even if vControl has stopped running or the connection to the instrument failed.*

11.9.28 Export Well Attributes

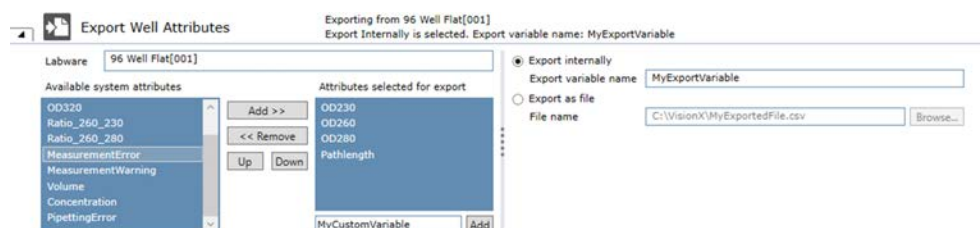


Fig. 126: Export Well Attributes

The **Export Well Attributes** command is used to export the well attribute values of a labware item during script runtime. The Export file is a (.csv) file, a semicolon (;) is used as a separator.

The header of the file contains the names of the exported well attributes. The file must contain a **Well Index** column. Each row below the header represents one well and its corresponding well attribute values.

Example

Well Index	OD230	OD260	OD280	PathLength
1	1.44	12.33	13.55	5.32
2	2.54	33.22	14.43	5.65

The raw text file of this example would then be:

```
Well Index;OD230;OD260;OD280;PathLength
1;1.44;12.33;13.55;5.32
2;2.54;33.22;14.43;5.65
```

Parameters

Labware: This will enable the user to select a labware from the worktable, from which the well attributes shall be exported.

Available System Attributes: This list contains all predefined well attributes that are registered in the system for export.

Attributes Selected for export: This list contains all the well attributes that shall be exported from the selected labware.

Custom Well Attribute Field: This field can be used to add custom attributes, which are not predefined in the system to the list of attributes selected for export. Clicking the **Add** button next to this field will add the custom well attribute to the list above.

Export Options: The export well attribute command contains two ways to export the selected well attributes.

- **Export Internally:** The generated export file is created by the system and is not meant to be edited by the user. This option should be used, if the user wishes to import the well attributes again during the same script runtime. Here, the user can define a unique export variable name which will register the exported well attributes to this variable. The user can then use the Import Well Attributes command (See Import well attributes for more details), enter the same export variable name, and import these attributes again to a different plate. Editing this file will cause the script execution to fail. If the user wishes to edit the export file, they should use the **Export as file** option.
- **Export as file:** The user can select a file using the **browse** button. This will then export the selected well attributes to the given location. This file will not be protected with a hash file and can be edited in between the export and import options. This option should also be used if the export and import commands are in separate script runtimes.

11.9.29 Import Well Attributes



Fig. 127: Import Well Attributes

The **Import Well Attributes** command is used to import well attribute values to a labware item during script runtime. The well attributes from the import file will be written to their corresponding well index of on the labware. The Import/Export functionality supports different labware types. Wells that are not available on the import labware are skipped. The import file must be a (.csv) file, a semicolon (;) must be used as a separator.

The header of the file contains the names of the exported well attributes. The file must contain a **Well Index** column. Each row below the header represents one well and its corresponding well attribute values.

Example

Well Index	OD230	OD260	OD280	PathLength
1	1.44	12.33	13.55	5.32
2	2.54	33.22	14.43	5.65

The raw text file of this example would then be:

```
Well Index;OD230;OD260;OD280;PathLength
1;1.44;12.33;13.55;5.32
2;2.54;33.22;14.43;5.65
```

Parameters

Labware: This will enable the user to select a labware from the worktable, to which the well attributes shall be imported.

Available System Attributes: This list contains all predefined well attributes, which are registered in the system for import.

Attributes Selected for import: This list contains all the well attributes that shall be imported to the selected labware.

Custom Well Attribute Field: This field can be used to add custom attributes, which are not predefined in the system, to the list of attributes selected for import. Clicking the **Add** button next to this field will add the custom well attribute to the list above.

Import Options: The import well attribute command contains two ways to import the selected well attributes.

- **Import Internally:** The User can define a variable name that will import the well attributes exported in a previous Export Well Attributes command. For more information on this option, see the **Export Well Attributes** command.
- **Import from file:** The user can select a (.csv) file, which contains the well attributes they wish to import. The file must contain a header, which describes the well attribute names. For more information on this option, see the **Export Well Attributes** command.

11.9.30 Normalization Calculation

The **Normalization Calculation** command calculates how much sample and diluent volume is needed to reach a specific target concentration and volume. The command supports the user to generate a script for a normalization workflow. The command should be used together with the **Transfer Individual Volumes** command, which can read the output files of the **Normalization Calculation** command and transfer the calculated sample and diluent volume to a plate. For more information on this, see **Transfer Individual Volumes**.

The **Normalization Calculation** command will create three CSV files as output. One file contains the pipetting instructions for the sample labware. The second file contains the pipetting instructions for the diluent labware, and the last file contains a plate map confirmation, which can be used by the **Plate Map Confirmation** command to show which wells are going to be normalized.

Normalization Calculation

Sample labware: '96 Well Flat[001]' Diluent labware: '100ml_1'
Optimal concentration: '20' Optimal volume: '30 [μl]'

Sample Labware: 96 Well Flat[001]
Diluent Labware: 100ml_1

Optimize for:
☒ Concentration
☐ Volume

Optimal values:
 Concentration: 20
 Volume: 30 [μl]

Concentration ranges:
 Minimum: 10
 Maximum: 30

Volume ranges:
 Minimum: 10 [μl]
 Maximum: 45 [μl]

Minimum pipettable sample volume: 1 [μl]
 Minimum pipettable diluent volume: 1 [μl]
 Sample well offset:
 Sample well range:
☐ Exclude wells with measurement warning
☐ Close gaps (applies to whole sample labware)

If the volume and concentration are out of the specified range, no normalization data will be calculated.

Fig. 128: Normalization Calculation

Parameters

Sample Labware: This field enables the user to select labware that contains the samples.

Diluent Labware: This field enables the user to select labware that contains the diluent.

Optimize for:

- **Concentration:** This selection enables an algorithm that optimizes the normalization according to the concentration. First, the algorithm tries to achieve both optimal values for concentration and for volume. In a second step, if optimal concentration and volume cannot be reached, the algorithm tries to achieve the optimal concentration, within the minimum and maximum volume ranges. If necessary in a third step, both volume and concentration are optimized in the allowed ranges, still prioritizing a result close to the optimal concentration. If no solution within allowed volume and concentration ranges is found the well will not be pipetted.
- **Volume:** This selection enables an algorithm that optimizes the normalization according to the volume. Similar to the previous optimization, both optimal values are tried to be met. In a second step, if optimal concentration and volume cannot be met, the algorithm tries to achieve the optimal volume within the minimum and maximum concentration ranges. If necessary in a third step, both volume and concentration are optimized in the allowed ranges, still prioritizing a result close to the optimal volume. If no solution within allowed volume and concentration ranges is found the well will not be pipetted.

Optimal Values: These fields contain the optimal values for concentration and volume. These are the main values, which the algorithm tries to achieve during normalization.

Concentration and Volume Ranges: These values define the range in which the algorithm can optimize the selected optimization setting. If the optimal values cannot be met, these ranges are taken into account, depending on the radio button selection.

Minimum pipettable sample and diluent volume: These fields define the minimum amount of sample or diluent which can be pipetted.

Exclude wells with measurement warnings: If a well contains a warning, which occurred during the measurement of the sample concentration, enabling this checkbox will exclude it from the normalization.

Sample well offset: Select by how many wells the starting point for the normalization calculation is shifted. This is usually expressed as a variable. No well offset will be applied, if this optional field is empty.

Sample well range: Select how many wells should be considered for the normalization calculation. This is usually expressed as a variable. Considering the shift by the well offset, all further wells of the sample labware will be considered for the normalization calculation, if this optional field is empty.

Close gaps: Enabling this checkbox removes all empty wells, which would occur during the normalization. This also alters the mapping between source and target labware. Close gaps applies to the whole sample labware and not only to a selected range, i.e. all wells up to the sample well offset are considered.

The command marks all wells, that have been considered for the normalization calculation, whether the normalization calculation has been successful or not.

11.9.31 In Situ Normalization Calculation

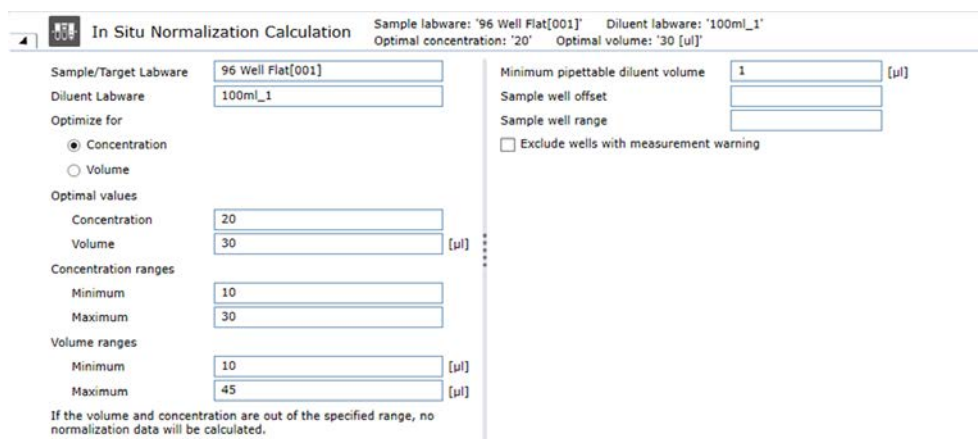


Fig. 129: In Situ Normalization Calculation

The **In Situ Normalization** calculation command performs a similar function to the Normalization Command. The difference between these commands is that the normalization is executed within the sample labware during an **In Situ Normalization Calculation**. This assumes that there is already a volume of sample in the target labware, since the sample and target labware are the same entity. Hence, it is not possible to add or remove a volume of sample from the wells of the source labware. It is only possible to add more diluent to the well. For more information on fields and algorithm of the command, refer to [“Normalization Calculation Command” \[260\]](#).

Sample well offset: Select by how many wells the starting point for the normalization calculation is shifted. This is usually expressed as a variable. No well offset will be applied, if this optional field is empty.

Sample well range: Select how many wells should be considered for the normalization calculation. This is usually expressed as a variable. Considering the shift by the well offset, all further wells of the sample labware will be considered for the normalization calculation, if this optional field is empty.

The command marks all wells, that have been considered for the normalization calculation, whether the normalization calculation has been successful or not.

11.9.32 Log

This command can be used to log additional information to the logfile. This can help to find relevant information later in the logfile.



Fig. 130: Log

11.9.33 Create Task

The Create Task command can be used to define tasks (Process iterations) to be executed by the system at a future time. One or more Create Task commands need to be followed by a Start Task command which applies the task definitions and stores them in the system's task repository.

To delay the execution of the process iterations, a date and time can be entered. The default of zero for both fields will start the new tasks as soon as possible. After selecting a Process, the list of labware groups is filled. For each entry (which correspond to the Source Steps of the selected Process), you need to specify a labware name or labware barcode. When executing the Process iteration, this labware will then be assigned to the Source Steps.

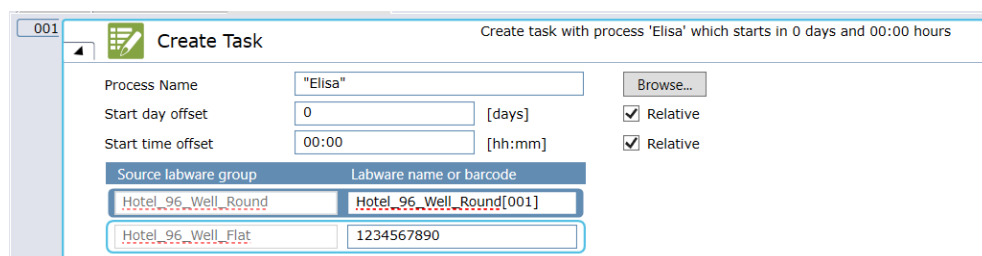


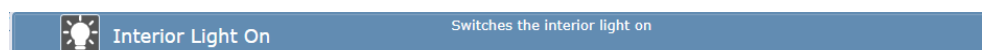
Fig. 131: Create Task

11.9.34 Start Tasks

Apply the tasks defined by preceding Create Task commands and store them in the system's task repository. See ["Create Task" ▶ 263](#).

11.9.35 Interior Light On

Switches the interior light on.



Also see about this

["Create Task" ▶ 263](#)

11.9.36 Interior Light Off

Switches the interior light off.



[Also see about this](#)

["Create Task" ▶ 263](#)

11.10 Touch Tools

Rich user prompts provide information (e.g., images, sound, worktable, loading information) that is shown as a prompt in Touch Tools during script or process runtime.

Rich user prompts can be shown in two different modes:

- **Display and Wait:** The script is paused and waits for the operator to touch **Continue**.
- **Display and Continue:** The script continues and runs in the background.

The Rich User Prompt software components are automatically installed when you install TouchTools and are visible as a new group Rich User Prompt with new script commands in the Controlbar.

11.10.1 Wizard Group

This command can be used to display a group of loading instructions to the operator at runtime. While the commands in this group are executed, the doors can be open.

Supported commands are:

TouchTools:	<ul style="list-style-type: none"> • all commands
Worktable:	<ul style="list-style-type: none"> • Add Labware • Remove Labware • Set Location • Register Labware
Sample-Tracking:	
Programming:	<ul style="list-style-type: none"> • Import Variable • Set Variable • Start Application • Start Vb • Log • Comment • End Script • Loop • Start loading command
: Loading ID	



Loop and if-else are not supported. Consider using TouchTools Wizard condition instead of 'if'.

If the wizard group contains only commands from the TouchTools group, it is possible to navigate forward and backward within the wizard.

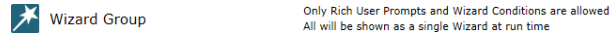


Fig. 132: Wizard Group

11.10.2 Standard (Rich User Prompt Display Command)

Use the Standard command to customize a new dialog to be shown on the touch monitor during script execution. Specify at least the Title and the Prompt Text. Press the Preview button to directly see your Rich User Prompt on the touch monitor.

Runtime behavior


During script execution, **Touch Tool** displays the **Rich User Prompt** template on the touch monitor user interface and displays the following:

- If parametrized, the picture is displayed.
- If parametrized, text is displayed. **Touch Tools** replaces a variable inside the text and XAML files by the corresponding variable value. Used functions are displayed using the total provided screen width and height.
- If parametrized, the sound is played with the configured amount of loops when the user prompt appears on the touch monitor.
- If parametrized, a progress bar is shown (optionally with percentage, remaining time, elapsed time, start time, and/or estimated end time value).

Parameters

The parameters of the Rich User Prompt (Display) command are as follows:

CAUTION! Specify at least the Title and the Prompt Text.


Standard

Progress Continue Show Progress Bar
Progress

Title

Progress

☐ Display and Wait
☒ Display and Continue

Picture File

...

Prompt Text

Progress

Sound File

...

Repeat sound

times

☒ Show progress bar

Time settings

☒ Display start time
☒ Display estimated end time

Progress settings

☐ Display time remaining
☐ Display time elapsed
☒ Display current percentage

Fix custom value

0

%

Progress bar elements

Start Time	Percentage %	End Time

☐ Change status light color

☐ Close prompt after

1

seconds

Preview Command on TouchScreen

Preview

Fig. 133: Standard Rich User Prompt

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Select Rich User Prompt mode:

- **Display and Wait:** The script is paused and waits for the operator to touch Continue.
- **Display and Continue:** The script continues and runs in the background.

Picture File: Optional file path to a picture. You can choose a still or moving image. Supported formats are: XAML, video (.wav), jpeg, gif, bmp, png

Convention:

- XAML files must be of type UserControl.
- XAML files must not contain the class Attribute.

- You can place Script variable placeholders inside XAML. Placeholders have the following convention: ~VariableName~

Prompt Text: Type the text to be displayed on the touch monitor. Those parts of your typed input which should be treated as text have to be enclosed in quotation marks (double-quotes). All parts in between will be considered as expressions. If the expressions cannot be resolved, the entire comment text is treated as plain text. If the File Path parameter is empty and there is a Text parameter, the font size will be larger.

Sound File: Choose an optional sound file to play.

Repeat sound: Specify the number of times to repeat the sound (no entry = endless loop).

Show progress bar: Display an optional progress bar. When you choose the required progress bar options, a schematic view of the progress bar design is also shown under **Progress bar elements**.

Time settings: You can select the Display start time and Display estimated end time check boxes.

Progress settings: You can select one of the following options: **Display time remaining**, **Display time elapsed**, or **Display current percentage**. In case of the Display current percentage option, you can type a value in the **Fix custom value** box in percent.

Progress bar elements: Automatic schematic view of the progress bar design reflecting your selected **Time settings** and Progress settings.

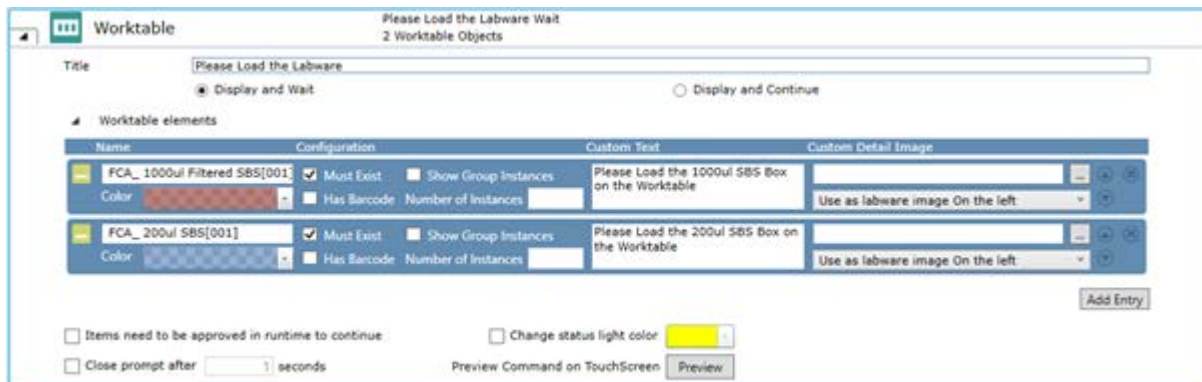
Change status light color: Select the color of the status lamp during the user prompt is shown from a color pallet.

Close prompt after: To close the dialog automatically after a certain time, select the **Close prompt after** check box and then type the number of seconds in the box.

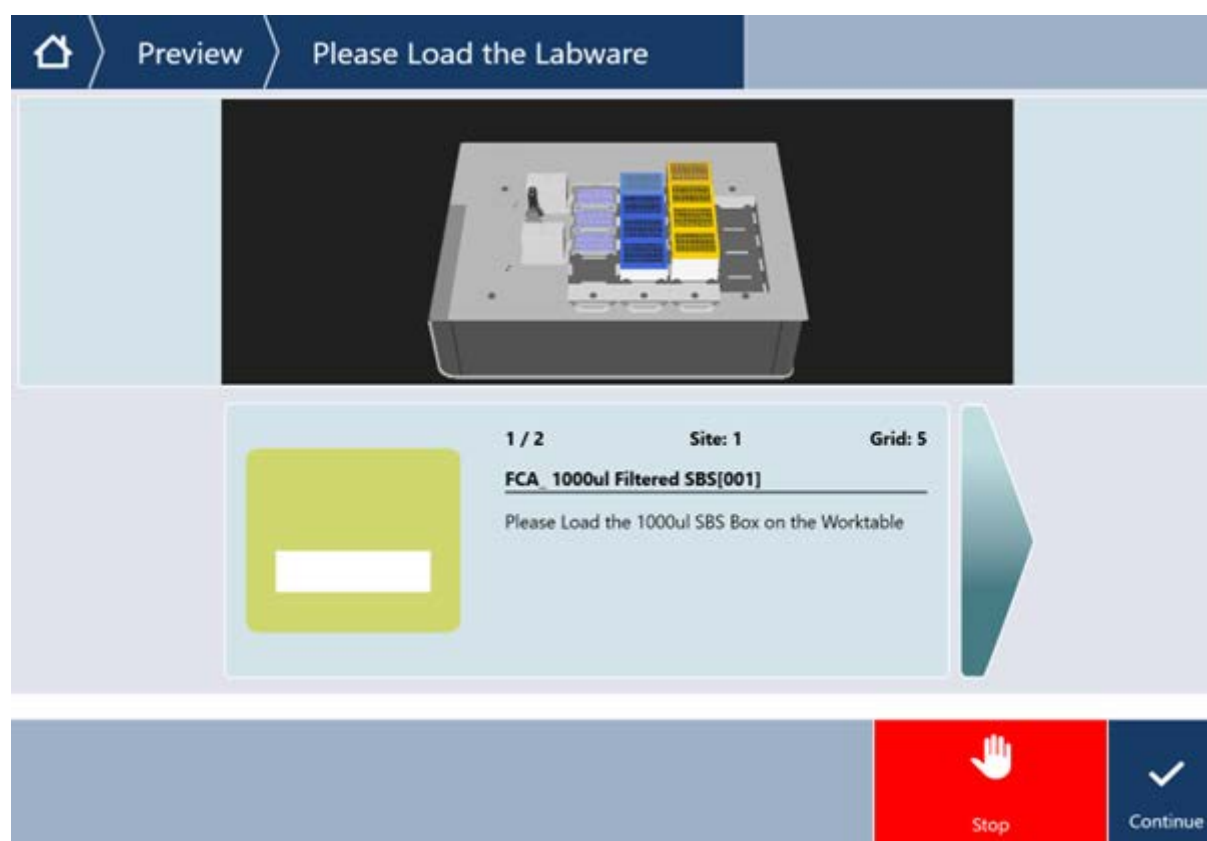
Preview: Click the **Preview** button to directly show your dialog on the touch monitor as it will appear at runtime. The colors you see in the preview may differ from the colors you see at runtime.

11.10.3 Worktable

The worktable command shows the operator a 3D visualization of the worktable with selectable carriers or labware blinking. A Text and Picture can be provided for each selected carrier to guide the operator through the worktable setup.



Name	Configuration	Custom Text	Custom Detail Image
FCA_1000ul Filtered SBS[001]	<input checked="" type="checkbox"/> Must Exist <input type="checkbox"/> Show Group Instances <input type="checkbox"/> Has Barcode Number of Instances: <input type="text"/>	Please Load the 1000ul SBS Box on the Worktable	Use as labware image On the left
FCA_200ul SBS[001]	<input checked="" type="checkbox"/> Must Exist <input type="checkbox"/> Show Group Instances <input type="checkbox"/> Has Barcode Number of Instances: <input type="text"/>	Please Load the 200ul SBS Box on the Worktable	Use as labware image On the left



Name: Selection of the labware/carrier to display. The labware name must be in a non-array variable.

Color: A color picker which defines the color in which the currently shown carrier or labware item should blink in on the TouchTools interface.

Must Exist: If the checkbox **Must Exist** is activated and a specified labware is not present during runtime on the worktable, an error message will be displayed. The operator has the choice to abort or ignore the entire prompt and continue. If the checkbox is deactivated, the Worktable prompt will always be shown and only the existing labware will be displayed in the TouchTools interface.

Show Group Instances: If the **Show Group Instances** checkbox is checked, labware items out of the labware group will be shown in touch tools. The field **Number of Instances** defines how many instances of the labware group shall be shown. The Selected labware item in the **Name** field defines the first labware item of the group which shall be shown in the TouchTools interface. For example, if 10 labware items in the same group are present on the worktable, the operator selected the third labware item of the group, and defines **Number of Instances** with the value 5, the labware items 3 to 7 will be displayed in the TouchTools interface.

Has Barcode: If the Checkbox Has Barcode is checked, a field to enter a barcode will be shown in the TouchTools interface. If the option Show Group Instances is selected, a barcode field will be shown for all the displayed labware items in the TouchTools interface.

Custom Text: Custom Text which can be displayed in the TouchTools interface when the carrier or labware item is shown.

Custom Detail Image: An image which can be displayed alongside of the Custom Text, replace the Custom Text entirely, or access an URI to show an image.

Items need to be approved in runtime to continue: This check box will show an extra Approve button in the TouchTools interface. Each item shown in the Worktable command will have to be approved before the command execution can be completed.

Close Prompt after X seconds: This will close the TouchTools user prompt after the defined number of seconds.

Change status light color: This will change the instrument status light color to a defined status light when the TouchTools interface is shown.

Preview Command on TouchScreen: This will show a preview of the command on the TouchScreen without needing to execute the entire script or method.

11.10.4 XAML

Use the XAML command to customize a new dialog from a XAML file.

11.10.5 Report

Use the Report command to generate reports from template files. The template which is used to convert from XAML to PDF must not contain a scrollviewer.

When the option "Convert XAML to PDF" is selected, the resulting PDF file contains bitmaps (BMP) of the rendered content, not plain text.

11.10.6 Close

Use the **Close** script command to force Touch Tools closing an open rich user prompt (of type Display and Continue) automatically.

11.10.7 Wait for Close

Use the Wait for Close command if a rich user prompt is displayed in Display and Continue mode to stop the script until the rich user prompt is either closed by the user or closed automatically by the timer.

11.10.8 Wizard Condition

If the condition in the Wizard Condition command is true, all Rich User Prompts within the Wizard Condition Group are shown on the touch monitor at run time.

11.10.9 Get File

If the condition in the **Wizard Condition** command is true, all Rich User Prompts within the Wizard Condition Group are shown on the touch screen at run time.

11.10.9.1 Edit Time

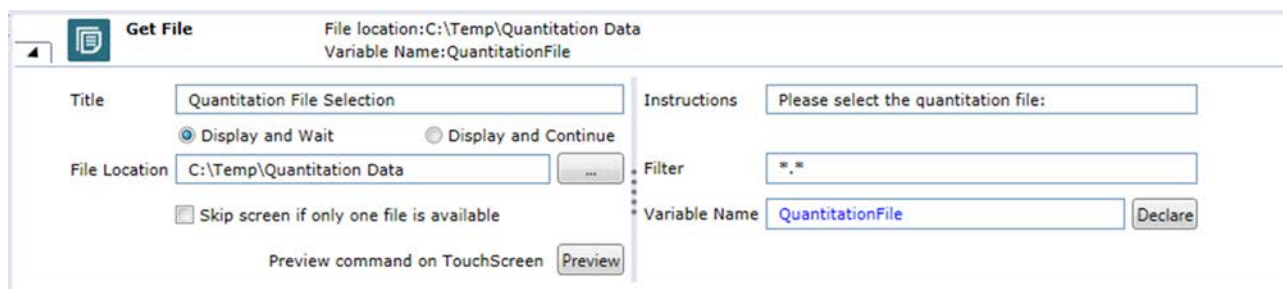


Fig. 134: Edit Time

Specify at least the **Title**, **File Location** and **Variable Name**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

File Location: Specify a folder of your choice where the selectable files are located. Depending on the defined Filter the files inside the defined folder will be displayed during script runtime.

Skip screen if only one file is available: If this option is set, the runtime screen will not be displayed during script runtime if there is only one file available within the defined folder and for the given filter.

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Filter: Specify a filter of your choice. Only the files inside the file location that fit the filter value will be displayed during script runtime.

Tab. 5: Example filters

Filter	Description
*	* is a wild card that filters for any combination of characters and any length string
.	Returns all files in a directory

Filter	Description
*.txt	Only files with the .txt extension will be displayed
QuantitationData.txt	There must be a file with the exact name within the folder, otherwise no file will be displayed.
A*.*	Only files starting with a filename starting with an "A" will be displayed.
.t	Only files with an extension that has a "t" as the first letter will be displayed.

Variable Name: Name of the variable where the file path and name (selected at runtime) will be stored. If the variable is not yet declared it is possible to declare the variable via the declare button.

Preview: Click the **Preview** button to display the prompt on the touchscreen as it will appear at runtime. The colors you see in the preview may differ from the colors used at runtime.

11.10.9.2 Runtime

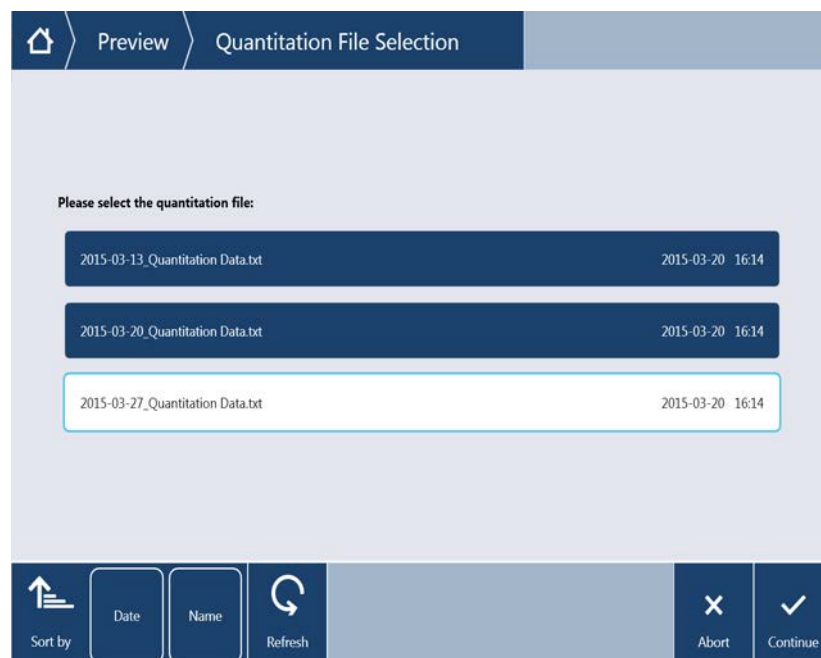


Fig. 135: Runtime

At runtime the touch screen displays a list of the files found inside the defined folder and filter.

Refresh: Will check the target directory again and update the selectable file names, if files that match the filter have been added or removed inside the defined folder.

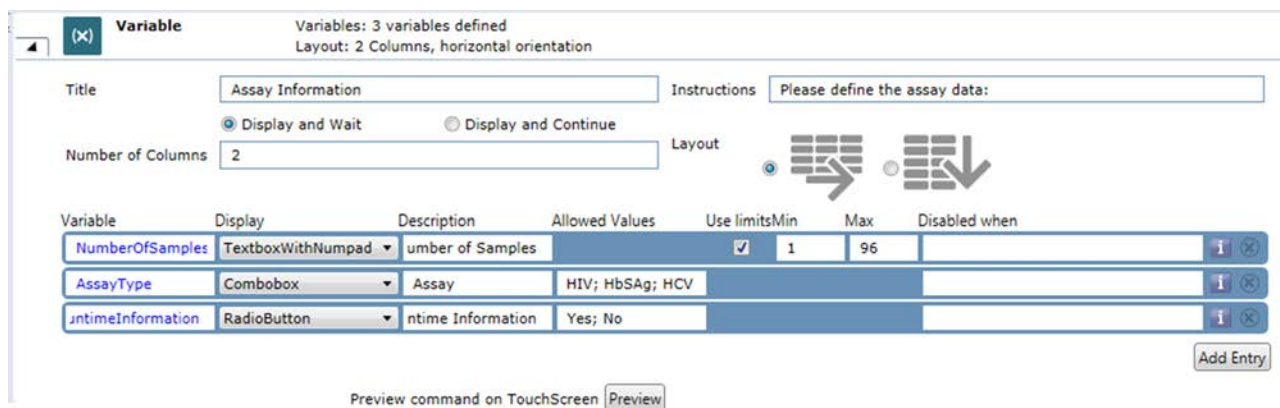
Abort: Will abort the entire run after operator confirmation.

Continue: Continues with the script.

11.10.10 Variable

The Variable Rich User Prompt allows defining a runtime screen to change variable values during script run. It is possible to assign to each declared variable a control type (e.g., Textbox, Combobox) which will be displayed inside the runtime screen in a defined order. At runtime the operator is able to enter / change / select values which be stored in the associated variable upon pressing continue.

11.10.10.1 Edit Time



Variable	Display	Description	Allowed Values	Use limitsMin	Max	Disabled when
NumberOfSamples	TextboxWithNumpad	umber of Samples		<input checked="" type="checkbox"/>	1	96
AssayType	Combobox	Assay	HIV; HbSAg; HCV			
untimeInformation	RadioButton	ntime Information	Yes; No			

Fig. 136: Edit Time

Specify at least the **Title**, **Number of Columns**, **Layout**, **Variable**, **Display** and **Description**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to press Continue.

Display and Continue: The script continues and runs in the background.

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Number of Columns: Defines the number of columns for the runtime screen to define its layout.

Layout: Defines the orientation of the controls within the runtime screen.

Variable: Specify the variable for which a runtime control shall be displayed.

Display: Specify the display runtime display type for the variable.

Tab. 6: Display Types

Display Type	Description
TextBox	Allows entering a text. When entering the textbox the TouchKeyboard will be opened automatically to be able to enter values via the Touch Screen.
TextBox, Numpad	Allows entering numbers. When entering the textbox the Numpad opens automatically to be able to enter values via the Touch Screen.
TextBox, No Keyboard	Allows entering text via the physical Keyboard or a Hand- held scanner. No Touch Support will be opened.
Checkbox	Allows enabling / disabling an option. Will store "True" or "False" to a string variable.
Dropdown	Allows selecting a pre-defined value from a list. Input the list
Radio Button	Allows switching between different options. Only one radio may be selected.
Empty	Placeholder to be able to adapt the layout of the screen. No control will be displayed at runtime at this position.

Description: Specify the header text for the control. For radio buttons this is the header for the full radio button group. For Checkboxes this is the label of the checkbox.

Allowed Values: Only visible for Dropdowns of Radio buttons. Keep in mind that the number of allowed values for radio button values influences the screen layout.

Specify the values contained inside the dropdown as a semicolon separated list. Specify the number of radio buttons and their label one group via a semicolon separated list.

Example for both combobox: OptionA;OptionB;OptionC

Use limit: Only visible for Textboxes with Numpad. Specify if limits shall be checked for the field.

Min / Max: Minimum and maximum limits for the field that will be checked at runtime.

Disabled when: Specify an expression that defines when the field shall be disabled. The expression is resolved before the screen is displayed. A change of a screen value will have no effect on this function.

Add Entry: Adds a new row to the list. Remove rows by clicking the (X) button to the right of the row you would like to remove.

11.10.10.2 Runtime

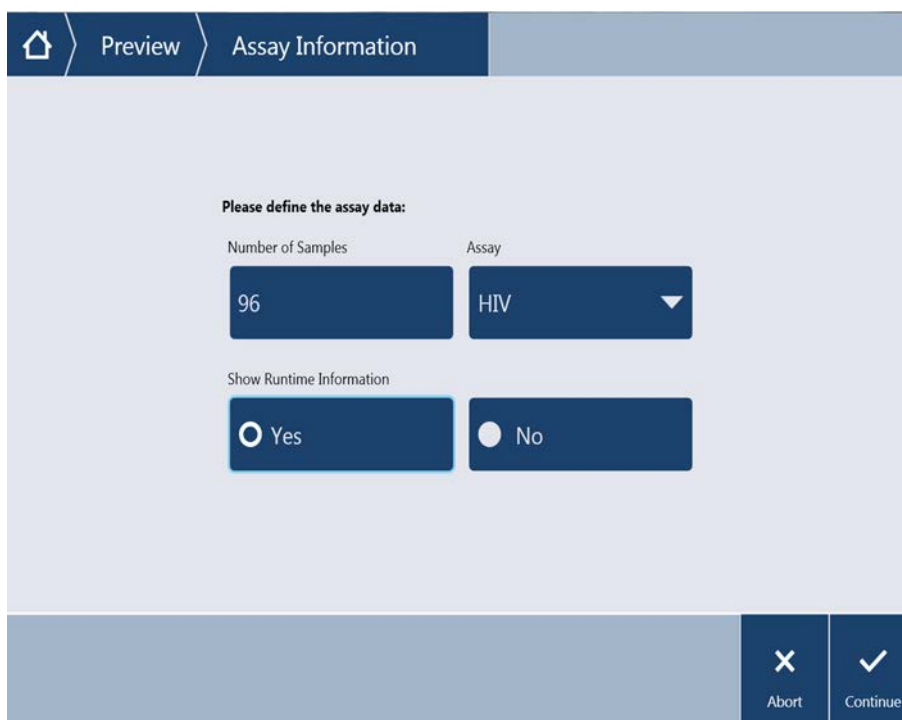


Fig. 137: Runtime

At runtime the touch screen displays the defined controls for the variables according to the defined layout.

The changed / updated / selected values are stored into variables when the operator presses continue.

Abort: Will abort the entire run after operator confirmation.

Continue: Continues with the script.

11.10.11 Confirmation

The Confirmation Rich User Prompt allows defining a runtime screen containing a custom image and text / XAML. This can be used to be displayed before instrument actions start and user confirmation is required, for example worktable loading. The screen can be customized to contain information that helps the user to be able to confirm the current status before continuing.

11.10.11.1 Edit Time

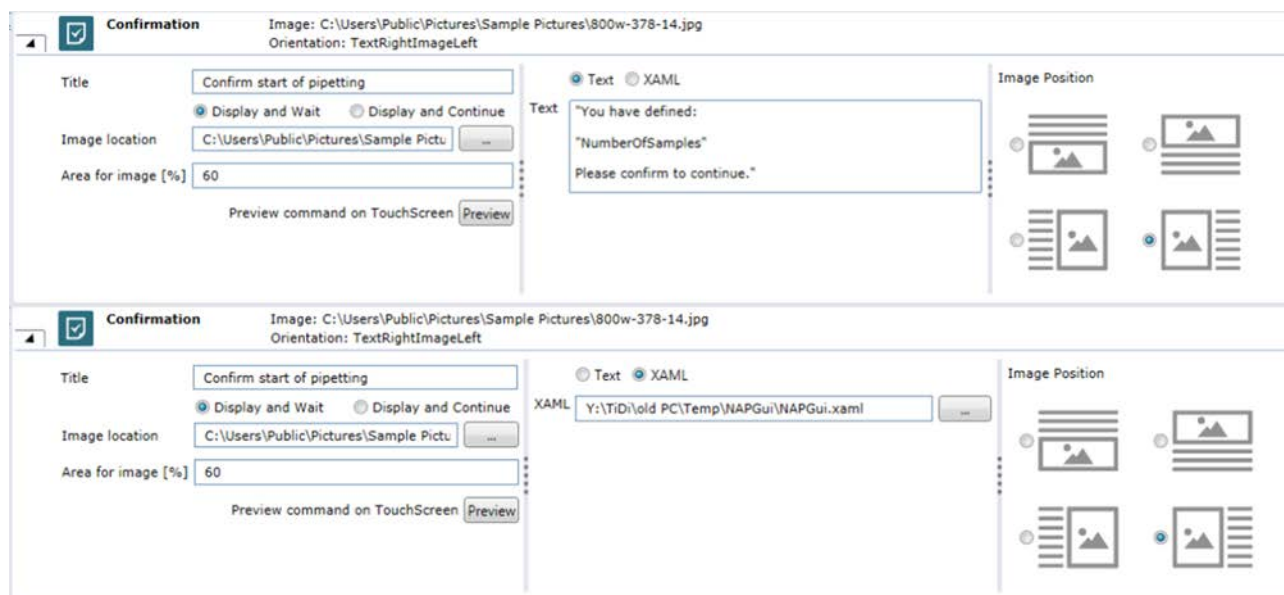


Fig. 138: Edit Time (Text Selected and XAML selected)

Specify at least the **Title**, **Image location**, **Area for image**, **Text** or **XAML**, and **Image Position**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

Image location: Specify the path to the image that shall be displayed at runtime at the defined position.

Area for image: Specify the available space to display the image in percent compared the space used by the text or XAML content. If the area is zero, no image will be displayed.

Text: Specify a custom text that will be displayed at runtime at the defined position. The text allows capture returns and Variables. When using variables the non-variable text has to be within quotation marks.

XAML: XAML is an Extensible Application Markup language developed by Microsoft, allowing defining user interfaces. With XAML it is possible to provide formatted text and images. Specify the path to a valid XAML file that shall be displayed at runtime. This XAML file does not support the binding functions of the XAML Rich User Prompt and just supports standard XAML without Veya specific bindings.

Example:

```
<UserControl xmlns=http://schemas.microsoft.com/winfx/2006/
xaml/presentation xmlns:x="http://schemas.microsoft.com/
winfx/2006/xaml">
    <Grid>
```

```
<Grid.RowDefinitions>
    <RowDefinition />
    <RowDefinition />
</Grid.RowDefinitions>
<TextBlock Grid.Row="0" FontSize="18">
    You have loaded 96 samples
</TextBlock>
<TextBlock Grid.Row="1" FontWeight="Bold"
Margin="0,20,0,0">
    Please confirm to continue
</TextBlock>
</Grid>
</UserControl>
```

Image Position: Specify the position for the image inside the runtime screen.

11.10.11.2 Runtime



Fig. 139: Runtime

At runtime the touch screen displays defined image / text / xaml in the defined size and orientation.

Abort: Will abort the entire run after operator confirmation.

Confirm: Continues with the script and logs the decision.

11.10.12 Plate Map Confirmation

The Plate Map Confirmation Rich User Prompt allows defining a runtime screen containing a plate or tube with well information. This can be used to be displayed before instrument actions start and user confirmation is required. The screen can be customized to contain information that helps the user to be able to confirm the following actions.

11.10.12.1 Edit Time

The figure shows two instances of the 'Plate map confir...' dialog box. The top instance has 'Labware Type' set to 'WellPlate96' and 'Number of Racks' is not visible. The bottom instance has 'Labware Type' set to 'TubeRunner' and 'Number of Racks' set to '1'. Both instances show fields for Title, File Location, Instructions, and a table for Column/Content mapping.

Fig. 140: Edit Time (Text Selected and XAML selected)

Specify at least the **Title**, **FileLocation**, **Labware Type**, **Well Number** and **WellInformation**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

File location: Specify the path to the file containing the well information. The file must be a semicolon separated CSV file and must contain at least the well number and the well information—e.g.:

```
SamplePlate.csv - Notepad
File Edit Format View Help
0;1;Content:\n1\ Sample1\n12\ MasterMix
1;2;Content:\n1\ Sample2\n12\ MasterMix
2;3;Content:\n1\ Sample3\n12\ MasterMix
```

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Labware Type: Specify the type of labware that shall be displayed at runtime. Possible selections are:

- 96 Well Plate
- 384 Well Plate
- Tuberunner

Number of racks: Specify the number of tuberunners that shall be displayed at runtime.

Column association: Associate the columns of the file (e.g., A, B, C) to the identifiers.

Well number: 1-based number referring to the well on the plate. It is counted from top to bottom and then left to right.

A1 = 1, A2 = 9... , B1 = 2, B2 =10

Well information: text allowing text formatting via for example \n. The text is displayed at runtime for the selected well.

ID: text that will be displayed at runtime within the well for an easier well identification. This value is optional.

11.10.12.2 Runtime

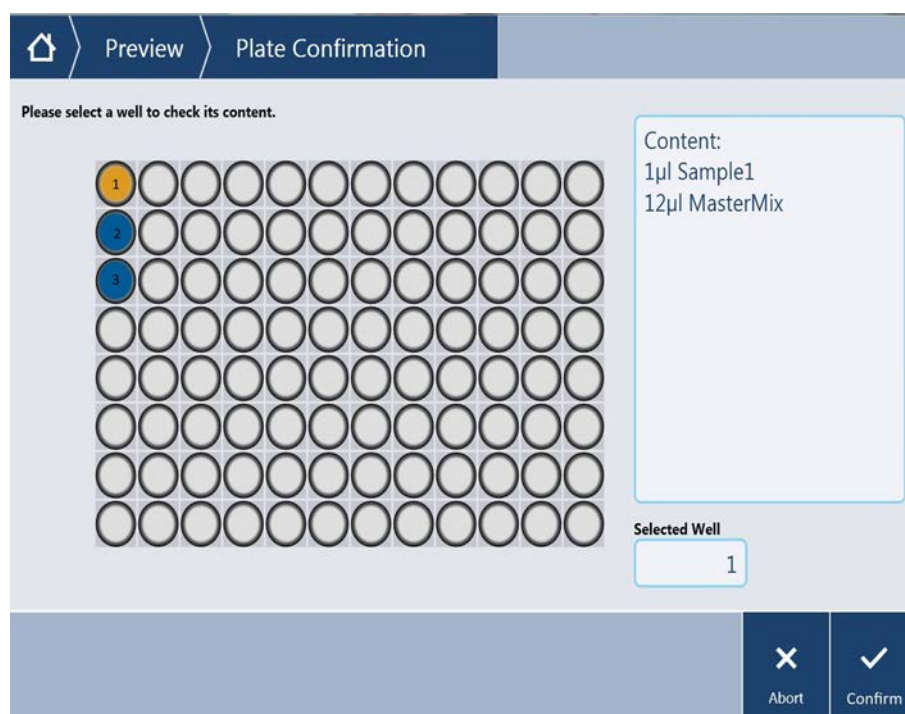


Fig. 141: Runtime 96-well plate

At runtime the touch screen displays the defined labware and well information. Empty wells are displayed gray.

Abort: Will abort the entire run after operator confirmation.

Confirm: Continues with the script and logs the decision.

11.10.13 Select Wells

The Select wells Rich User Prompt allows selecting the number of samples for the current run by defining the start and end well of a pre-defined labware.

There are different selection options that can be defined for the runtime screen.

On continue the selected values are stored into defined variables and can be used for the subsequent script.

11.10.13.1 Edit Time

The screenshot displays two configuration panels for the 'Select Wells' command. The top panel is for 'WellPlate96' in 'RowBased' mode. It includes fields for 'Title' (Define the sample number), 'Labware Type' (WellPlate96), 'Mode' (RowBased), 'Start Well' (StartWell), and 'Sample Count' (NumberOfSam). The bottom panel is for 'TubeRunner' in 'FreeRectangle' mode. It includes fields for 'Title' (Define the sample number), 'Labware Type' (TubeRunner), 'Mode' (FreeRectangle), 'Number of racks' (2), 'Start Well' (StartWell), and 'End Well' (EndWell). Both panels have 'Display and Wait' selected, an instruction field, and a 'Preview' button.

Fig. 142: Edit Time (Well Plate and Tube Runner selected and different modes.)

Specify at least the **Title**, **LabwareType**, **Number of racks**, **Mode**, and **Variables**.

Title: Specify a title of your choice. The title is shown above the Rich User Prompt during script runtime.

Display and Wait: The script is paused and waits for the operator to touch Continue.

Display and Continue: The script continues and runs in the background.

Instructions: Specify an instruction text of your choice. The instruction text is shown above the file list during script runtime.

Labware Type: Specify the type of labware that shall be displayed at runtime. Possible selections are:

- 96 Well Plate
- 384 Well Plate
- Tuberunner

Number of racks: Specify the number of tube runners that shall be displayed at runtime.

Mode: Specify the well selection mode of the runtime screen. For each mode at runtime you have to select the start and the end well. The wells will be selected according to the defined mode.

Rowbased: selection starts at the selected start well and continues row-wise until the selected end well.

Columnbased: selection starts at the selected start well and continues column-wise until the selected end well.

Free Rectangle: selection starts at the selected start well and continues to the end well and builds a rectangle.

Start Well: Specify the variable where the selected start well number will be stored (1 – based) after continuing at runtime.

End Well: Specify the variable where the selected end well number will be stored (1 – based) after continuing at runtime.

Sample Count: Specify the variable where the number of selected samples will be stored after continuing at runtime.

11.10.13.2 Runtime

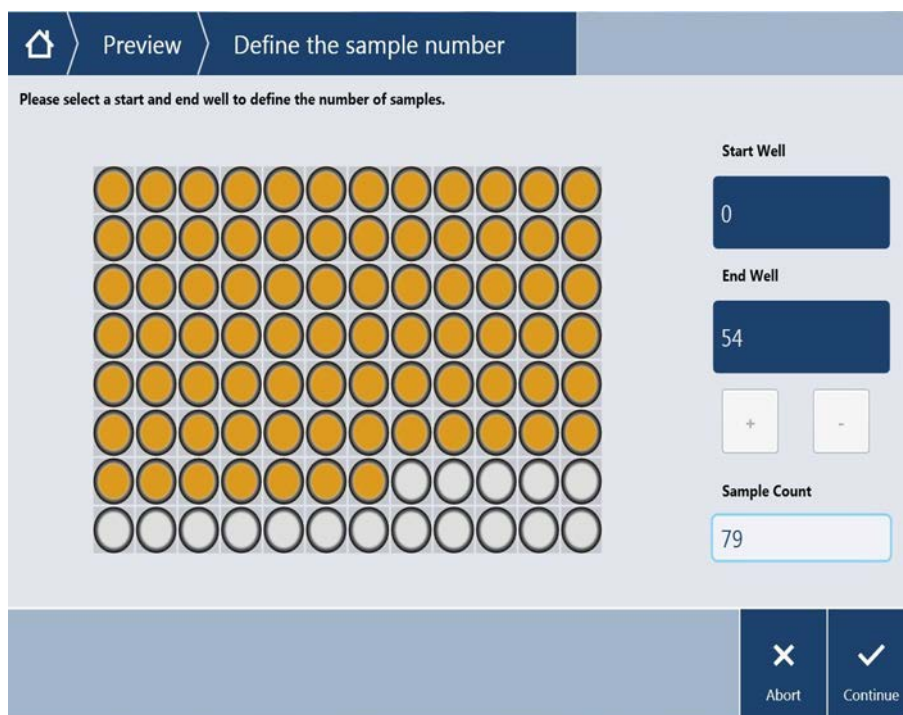


Fig. 143: Runtime

At runtime the touch screen displays the defined labware type and allows selecting the start and end well. Wells are selected automatically according to the defined mode.

Start Well: Specify the start well by either clicking on a well on the labware or using the + and – buttons. If no well is selected yet the first selection will be the start well.

End Well: Specify the end well by either clicking on a well on the labware or using the + and – buttons. If one well is already selected the next selection will be the end well.

Abort: Will abort the entire run after operator confirmation.

Confirm: Continues with the script and logs the decision.

11.10.14 Manually Scan Labware

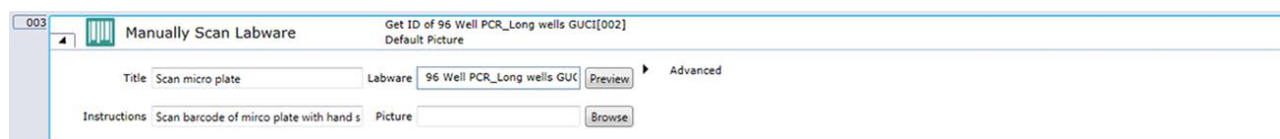


Fig. 144: Manually scan labware

Allows manually entering a barcode or identification to any labware during the run. It does not require a barcode scanner device on the deck. A barcode can be entered by handheld barcode scanner, touch monitor or key board. The software will prompt the user to manually enter a barcode for the specified labware. The command will then assign the entered barcode to the labware.

Title: Define a title for this command. The title is shown during the run on top of the touch monitor.

Instructions: Define instructions for the operator which is shown during the run on touch monitor.

Labware: Define the specific labware. This can be done by click on selected labware on the virtual worktable, entering the name or an expression.

Picture: If a picture is selected here it will be displayed during the run on touch monitor. If no picture is selected, the 3D worktable will be displayed.

Advanced

Fig. 145: Manually scan labware advanced

Duplicate IDs allowed: If this is checked duplicate IDs are allowed and operator will get no error message. If not checked, the operator will get an error message on the touch monitor about duplicate barcodes and their position.

Color: Define a custom color to highlight the labware in the default worktable graphic.

11.11 Worklist



Fig. 146: Worklist: Commands

NOTICE

Verify that the desired pipetting positions and results are achieved

When using worklist from either CSV or GWL files, please ensure the format is correct and verify that the desired pipetting positions and results are achieved.

To execute a worklist in a script, use the **Load Worklist** command to load the worklist file and specify additional parameters for the pipetting operations. Then insert an **Execute Worklist** command to run the worklist file. You can load several worklists by inserting several load worklist commands in your script. In this case, a single **Execute Worklist** command runs all of the worklists which you have loaded. They will be executed in chronological order of loading. vControl keeps a track of the worklists which have been run, and a second Execute Worklist command will not run them again unless you have reloaded them.

11.11.1 Worklist File Format

A worklist file is a text file containing pipetting instructions.

The worklist commands consist of individual lines (or records). Seven different record types are available: **Aspirate**, **Dispense**, **Wash Tips / Replace DiTis**, **Flush**, **Break**, **Set DiTi Type** and **Comment**. Each record starts with a single character indicating the type and is followed by one or more parameters. The parameters are separated by semicolons. Some of the parameters are optional and some record types have no parameters.

Aspirate: The Aspirate record has the following structure:

```
A; RackLabel; RackID; RackType; Position; TubeID; Volume; Liquid-
Class; Tip Type ; TipMask; ForcedRackType
```

Each Aspirate record specifies the aspiration parameters for a single tip (the next unused tip from the tip selection you have specified).

Dispense: The Dispense record has the following structure:

```
D; RackLabel; RackID; RackType; Position; TubeID; Volume; Liquid-
Class; Tip Type
; TipMask; ForcedRackType
```

Each Dispense record specifies the dispensing parameters for a single tip. It uses the same tip which was used by the preceding Aspirate record.

Tab. 7: Parameters for Aspirate and Dispense records in a worklist file

Parameter	Range	Explanation
RackLabel	Max. 32 characters	User-defined label (name) which is assigned to the labware.
RackID	Max. 32 characters	Labware barcode.
RackType	Max. 32 characters	Labware type (configuration name)—e.g.: “384 Well, landscape”.
Position	1 .. number of wells	Well position in the labware. The position starts with 1 and increases from rear to front and left to right.
TubeID	Max. 32 characters	Tube barcode.
Volume	0 .. +7158278	Pipetting volume in µl.
LiquidClass	Max. 32 characters	This optional parameter overwrites the liquid class specified in the Worklist command.
TipType		Reserved, must be omitted.
TipMask	1 .. 128	This optional parameter specifies the tip you want to use. The tip number is bit-coded (i.e., Tip 1 = 1, Tip 2 = 2, Tip 3 = 4, Tip 4 = 8, ...). Please note that you can only use a tip which has been enabled with Tip Selection in the Worklist command.
ForcedRack- Type	Max. 32 characters	This optional parameter is ignored by vControl.
MinDetected- Vol- ume	0 .. +7158278	This optional parameter is ignored by vControl.

Wash Tip/Replace DiTi: The Wash Tip/Replace DiTi record has the following structure:

Tab. 8: Parameters for wash tip/replace DiTi

Parameter	Explanation
W;	To use the first wash scheme in the worklist command

The Wash Tip / Replace DiTi record washes the tip (or replaces the DiTi) which was used by the preceding Aspirate record. There are no parameters (the wash position, wash parameters and/or drop DiTis position are specified in the Worklist command).

Decontamination Wash: The Decontamination Wash record has the following structure:

Tab. 9: Parameter for decontamination wash

Parameter	Explanation
WD;	vControl converts the 'WD' command into an Aspirate (of the decontamination liquid specified in the Load Worklist command) followed by a wait and then a wash.

Decontamination wash consists of a decontamination wash operation followed by the normal wash procedure. There are no parameters (the parameters are specified in the Worklist command).

Flush: The Flush record has the following structure:

Tab. 10: Parameter for flushing

Parameter	Explanation
F;	It discards the contents of the tips without washing them or dropping DiTis.

Break: It has the following structure:

Tab. 11: Parameter for break

Parameter	Explanation
B;	The Break record forces the execution of previously specified aspirate, dispense or wash actions which have not yet been executed.

If you don't specify a Break record, vControl normally executes pipetting commands in groups to optimize the efficiency. For example, if you have specified four tips in the Worklist command, vControl will queue Aspirate records until four of them are ready for execution. This allows pipetting to take place using all four tips at the same time. Specify the Break record if you want to execute all of the

currently queued commands without waiting. You can use the Break record—for example, to create a worklist that pipettes using only one tip at a time (even if you chose more than one tip in the tip selection).

Set DiTi Type: It has the following structure:

```
S;DiTi_Index
```

When using DiTis, S; is used to switch DiTi types from within a worklist. Choose the required DiTi type by specifying the DiTi index. vControl automatically assigns a unique index to each DiTi type. The DiTi index is shown in the **General Settings** tab of the DiTi labware.

If the S; record is not used, the worklist uses the DiTi type which you select with the Get DiTi button in the Worklist command.

The Set DiTi Type record can only be used at the very beginning of the worklist or directly after a Break record. A Break record always resets the DiTi type to the type selected in the Worklist command. Accordingly, if your worklist contains a Break record, you may need to specify the Set DiTi Type record again.

Comment: The Comment record has the following structure:

```
C;Comment
```

Start Timer: Start a timer. The Start Timer record has the following structure:

```
TS;TimerNumber
```

Wait for Timer: Wait for a defined number of seconds after starting the timer started with a previously executed TS; record. The Wait for Timer record has the following structure:

```
TW;TimerNumber;WaitTimeInSeconds
```

Start Timer and Wait for Timer records must not be used within a pipetting sequence using A; and D; records.

Reagent Distribution: The Reagent Distribution record has the following structure:

```
R;AspirateParameters;DispenseParameters;Volume;Liquid-  
Class;NoOfDit iRe uses;NoOfMultiDisp;Direction[;ExcludeDest-  
Well]*
```

where:

```
AspirateParameters =  
SrcRackLabel;SrcRackID;SrcRackType;SrcPosStart;SrcPosEnd;
```

and:

```
DispenseParameters =  
DestRackLabel;DestRackID;DestRackType;DestPosStart;Dest-  
PosEnd;
```

Tab. 12: Aspirate Parameters for the Reagent Distribution command

Parameter	Range	Explanation
SrcRackLabel	Max. 32 characters	User-defined label (name) that is assigned to the source labware.
SrcRackID	Max. 32 characters	Source labware barcode.

Parameter	Range	Explanation
SrcRackType	Max. 32 characters	Source labware type (configuration name)—e.g.: "384 Well, landscape".
SrcPosStart	1 .. number of wells	First well to be used in the source labware.
SrcPosEnd	1 .. number of wells	Last well to be used in the source labware.

Tab. 13: Dispense Parameters for the Reagent Distribution command

Parameter	Range	Explanation
DestRackLabel	Max. 32 characters	User-defined label (name) that is assigned to the destination labware.
DestRackID	Max. 32 characters	Destination labware barcode.
DestRackType	Max. 32 characters	Destination labware type (configuration name)—e.g.: "384 Well, landscape".
DestPosStart	1 .. number of wells	First well to be used in the destination labware.
DestPosEnd	1 .. number of wells	Last well to be used in the destination labware.

Tab. 14: Additional Parameters for the Reagent Distribution command

Parameter	Range	Explanation
Volume	0 .. +7158278	Dispense volume in the destination labware in µl.
LiquidClass	Max. 32 characters	This optional parameter overwrites the liquid class specified in the Worklist command.
NoOfDitiReuses	-	Optional maximum number of DiTi reuses allowed (default 1 = no DiTi reuse).

Parameter	Range	Explanation
NoOfMultiDisp	-	Optional maximum number of dispenses in a multidispense sequence (default 1 = no multi-dispense).
Direction	-	Optional pipetting direction (0 = left to right, 1 = right to left; default = 0).
[ExcludeDestWell] *		Optional list of wells in destination labware to be excluded from pipetting.

The Reagent Distribution record specifies the aspiration and dispensing parameters for all of the tips you have chosen (and dispenses the same volume in all of the chosen wells in the destination labware).

The Reagent Distribution record has the following advantages when compared to separate Aspirate and Dispense records:

- Pipetting is automatically optimized during script runtime for the number of tips which are actually fitted to the FCA.

Sample Transfer: The Sample Transfer record has the following structure:

```
T;AspirateParameters;DispenseParameters;Volume;LiquidClass;NoOfDiTiReuses;NoOfMultiDisp;SampleCount;ReplicationCount;SampleDirection;ReplicateDirection[;ExcludeDestWell]*
```

where :

```
AspirateParameters =
SrcRackLabel;SrcRackID;SrcRackType;SrcPosStart;SrcPosEnd;
and: DispenseParameters =
DestRackLabel;DestRackID;DestRackType;DestPosStart;DestPosEnd
;
```

and:

```
DispenseParameters =
DestRackLabel;DestRackID;DestRackType;DestPosStart;DestPosEnd
;
```

Worklist Examples Using Labware Labels

Use this worklist format if you want to use labware labels to specify source and/or destination positions in the worklist. This is necessary if your labware does not carry barcodes.

In this case, you only need to define the following parameters in an Aspirate or Dispense record:

```
A;RackLabel;;RackType;Position;;VolumeD;RackLabel;;RackType;Position;;Volume
```

Parameter **RackType** is optional. If specified, vControl checks whether the labware specified with **RackLabel** has the configuration name specified with **RackType**. An error message is displayed if an incorrect labware type is found on the Veya worktable.

Parameter **ForcedRackType** is optional. If specified, vControl does not use the labware type of the labware specified with **RackLabel**, but uses **ForcedRackType** instead. The following worklist example illustrates the use of **ForcedRackType** in an **Aspirate** and a **Dispense** command:

```
A;S1;;;9;;10;;;Microplate, landscapeD;D1;;;1;;10;;;Deep-  
well, landscape
```

landscape If you specify the **ForcedRackType** parameter, you must make sure that labware of the required types have been placed in the right positions on the instrument's worktable before the worklist is executed.

The system liquid is associated with a "virtual" labware item of type **System** and labware label **Systemliquid**. The following worklist command can be used to aspirate the system liquid:

```
A;Systemliquid;;System;1;;500:Water
```

The following worklist example aspirates 10.1 µl from well A2 in labware "S1", dispenses 6 µl into well A1 in labware "D1", dispenses 4 µl into well B1 in labware "D1" and then washes the tip / changes the DiTi:

```
A;S1;;Microplate, landscape;9;;10.1D;D1;;Microplate, land-  
scape;1;;6D;D1;;Microplate, landscape;2;;4W;
```



Note that vControl counts wells from rear to front and then from left to right, and a Microplate, landscape labware has eight wells in each column. Accordingly, well A2 is specified in the above worklist as "9".

11.11.2 Convert CSV to GWL (Worklist)

The **Convert CSV to GWL** command reads and parses a CSV file after parametrization and generates a corresponding GWL worklist file that can be loaded and executed.

The **Convert CSV to GWL** command is used to create a vControl worklist from an Excel file which is stored in CSV format.

Parameters

CSV file to import: Specify the file name and location of the CSV file to import. Enter the path and file name directly or click **Browse** and choose an existing file.

Start with line: Conversion normally starts with the first line of the import file. Specify a line number which is higher than 1 to skip one or more lines at the beginning of the file.

Stop with last line: Stop conversion with the last line of the import file.

Stop with line: Conversion normally stops with the last line of the import file. Specify the line number where you want the conversion to stop. The remaining lines in the import file are skipped.

Column Separator: Select from the list the separator which is used between the fields (columns) in each of the records (i.e. lines) of the import file. The default is **Comma**.

Generated Script (Subpath and Scriptname): Specify the file name and location of the worklist file (*.gwl) which should be created. Enter the path and file name directly or click **Browse** and choose an existing worklist file. The **Convert CSV to GWL** command overwrites the previous contents of the file each time it is executed.

Content: The **Convert CSV to GWL** command can handle up to 11 different input parameters in the import file (labware name, labware barcode, tube barcode, worktable position for both source and destination carrier; volume, liquid class, and tip). For each column (field) of the import file, specify the assignment to one of the 11 parameters by choosing from the drop-down list. Alternatively, leave the Content field empty to skip the field because it is empty or contains irrelevant data. An error message is output if you assign the same parameter to more than one column of the input file or make a meaningless or contradictory assignment.

11.11.3 Load Worklist

Use this command to parse a GWL worklist file for the selected FCA device and let vControl convert it into a subroutine script.

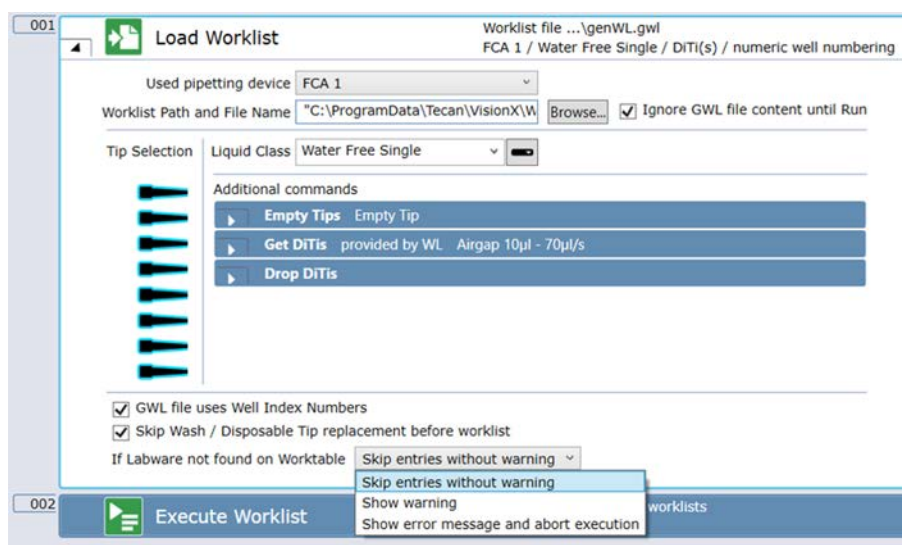


Fig. 147: Load Worklist

The **Load Worklist** command is used to load one or more worklist files, which can then be run with the **Execute Worklist** command. A worklist file is a text file containing pipetting instructions. It contains information on source and destination positions and the volumes to pipette. Worklists can be created using a plain text editor. They can also be created automatically by the **Convert CSV to GWL** command.


You can load several worklists by inserting several **Load Worklist** commands in your script. In this case, a single **Execute Worklist** command runs all the worklists which you have loaded. They will be executed in chronological order of loading.

vControl keeps a track of the worklists which have been run, and a second **Execute Worklist** command will not run them again unless you have reloaded them.

The **If Labware not found on Worktable** drop-down list is used to specify how to react if the worklist contains labware barcodes or labware labels of labware that is not located on carriers accessible to pipetting arm (as determined by the checkbox accessible to pipetting arm of the carrier).

- **Skip entries without warning:** skips any affected transfers from worklist.
- **Show warning** skips an affected worklist and shows the details of the skipped transfers in the trace view as warning. Run will finish with warnings and if electronic signatures are enabled the operator has to confirm these warnings at the end of the run.
- **Show error message** interrupts execution of worklist and shows an error message to the operator. The operator can choose to ignore and continue (run will finish with errors) or to abort execution of the script.

Additional Parameters


Load Worklist
Load a worklist file
FCA 1 / Water Free Single / DiTi(s) / numeric well numbering

Used pipetting device FCA 1

Worklist Path and File Name Browse...
☒ Ignore GWL file content until Run

Tip Selection

Liquid Class Water Free Single

Additional commands

Empty Tips

Empty Tip

Liquid Class

Empty Tip

Get DiTis

Airgap 10µl - 70µl/s

DiTi Types

Dynamic DiTi Handling

Airgap volume

10 µl

Airgap speed

70 µl/s

Drop DiTis

Drop location

☒ GWL file uses Well Index Numbers
☐ Skip Wash / Disposable Tip replacement before worklist

Fig. 148: Load Worklist: Additional Commands

Dynamic DiTi Handling: The dynamic DiTi handling is designed to use different DiTi types depending on the Aspirate volume. If you enter in the Dynamic DiTi Handling box—e.g.:

```
30:FCA DiTi 50µl;120:FCA DiTi 200µl;
```

this means that:

- DiTi 50 µl are used for volumes up to 30 µl,
- DiTi 200 are used for volumes > 30 to 120 µl, and
- for larger volumes the selected DiTi type from the Load Worklist command is used.

In the generated script, this leads – depending on which aspirates records got merged by vControl – to several **Get Tips** commands. When reusing tips, the tip type corresponding to the largest of the individual A;D; transfers preceding the next W; record is used.

Parameters

Used Pipetting Device: This parameter selects the FCA which is used to execute the worklist.

Worklist Path and File Name: Specify the file name and location of the worklist file (*.gwl). Enter the path and file name directly or click **Browse** and choose an existing file. This parameter accepts the name of a GWL text file in the form of an expression (this means, either a fixed value or script variables can be used to determine the real name of the file).

Ignore GWL file content until Run: Use this check box to indicate whether the GWL file contents shall be examined as part of a context check of the Load Worklist command or not (this is useful in such cases, when the GWL file is not dynamically created / modified during execution of the script). The default state for a new "Load Worklist" command for this check box is selected.

Tip Selection: Select the tips you want to use. If the worklist file specifies the tip selection (with the TipMask parameter) it overrides the settings you make here. If the commands in the corresponding worklist specifies a dedicated tip (with the TipMask parameter), this tip must be part of the tip mask set in the command, otherwise the system will report an error during parsing of the GWL file. It is not possible to use a mixed tip configuration (fixed tips and DiTis) in the same worklist. Therefore the system does not allow choosing fixed tips if DiTi tips have been already selected and vice versa.

Liquid Class: Select a liquid class from the list of currently known liquid classes to be used as default (i.e., if not explicitly specified in the GWL file) or use a variable. You can choose a different liquid class for each source labware and for the destination labware.

GWL file uses Well Index Numbers: Uncheck this box if the GWL file contains alphanumeric well numbers (A1, A2, ..., H11, H12) instead of numeric well numbers (1, 2, ..., 95, 96). The box is checked and numeric well numbers are expected by default.

Skip Wash / Disposable Tip replacement before worklist: By default, the execution of a worklist begins with a Wash (in case of fixed tips) or a DiTi replacement (Drop Tips followed by Get Tips, in case of DiTis). You can explicitly skip the default behavior of wash or DiTi replacement at the beginning of the worklist by selecting this check box. The check box is cleared by default.

11.11.4 Advanced Worklist

With an advanced worklist you can add any script command to your worklist. To do so, just add a Break command (**B;**) to your .gwl file followed by the deserialized script command that shall be executed.

Per Break command only one script statement may be specified. And the command must be encapsulated in a **ScriptGroup** element:

```
B;<ScriptGroup><Objects><Object Type="Command Type">Command
Serialization</Object></Objects><Name></
Name><IsBreakpoint>False</
IsBreakpoint><IsDisabledForExecution>False</
IsDisabledForExecution><LineNumber>0</LineNumber></
ScriptGroup>
```


Command Type and Command Serialization are specific to the command. The parameterization of the command is part of the command serialization.

Here is an example for a Transfer Labware command for RGA1 which moves labware "SourcePlate[001]" to **Nest7mm_Pos**:

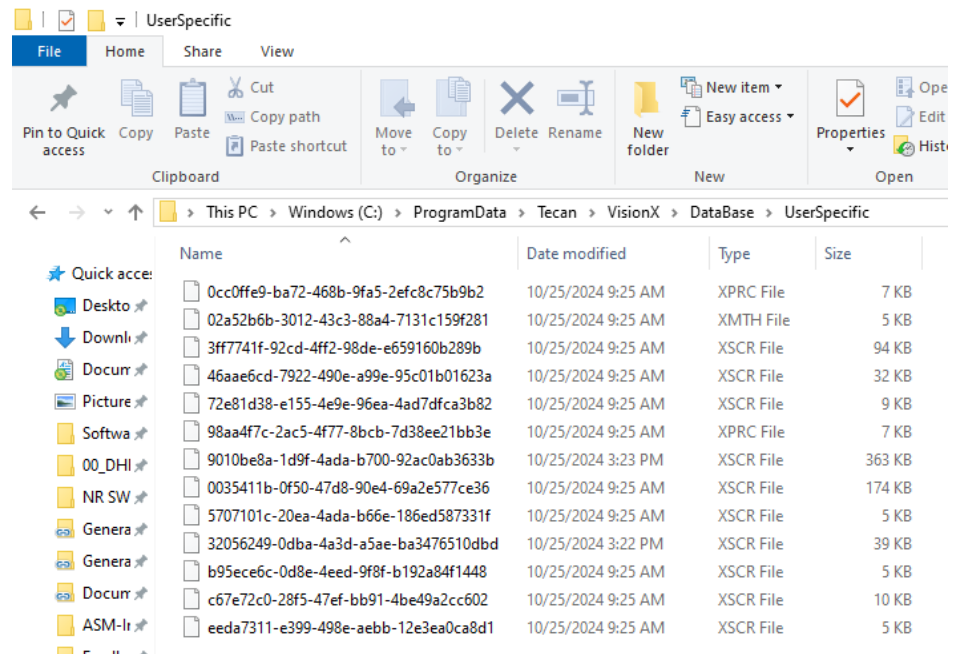
```
B;<ScriptGroup><Objects><Object
Type="Tecan.VisionX.ApplicationDriver.ApplicationDriverBase.A
pplicationDriverMacro"><ApplicationDriverMacro Version="1"
Name="RGA1_TransferLabware" ModuleName="RGA 1"
ExecutionTime="PT2S" IsBreakpoint="false"
IsDisabledForExecution="false"
LineNumber="1"><ExecutionSettings>&lt;TransferLabwareComm
andParameters xmlns:i="http://www.w3.org/2001/XMLSchema-
instance"
xmlns="http://schemas.datacontract.org/2004/07/
Tecan.VisionX.Drivers.RobotDriverBase"&gt;&lt;FixedSi
te&gt;&lt;false&gt;/
FixedSite&gt;&lt;Labware&gt;SourcePlate[001]&gt;
&lt;/
Labware&gt;&lt;Location&gt;Nest7mm_Pos&gt;/
Location&gt;&lt;MoveToBase&gt;false&gt;/
MoveToBase&gt;&lt;OnTheFlyTool&gt;&lt;/
OnTheFlyTool&gt;&lt;Site&gt;1&gt;/
Site&gt;&lt;UseOnTheFlyTool&gt;false&gt;/
UseOnTheFlyTool&gt;&lt;/
TransferLabwareCommandParameters&gt;</
ExecutionSettings></ApplicationDriverMacro></Object></
Objects><Name></Name><IsBreakpoint>False</
IsBreakpoint><IsDisabledForExecution>False</
IsDisabledForExecution><LineNumber>0</LineNumber></
ScriptGroup>
```

There is an easy way to get the needed serialization:

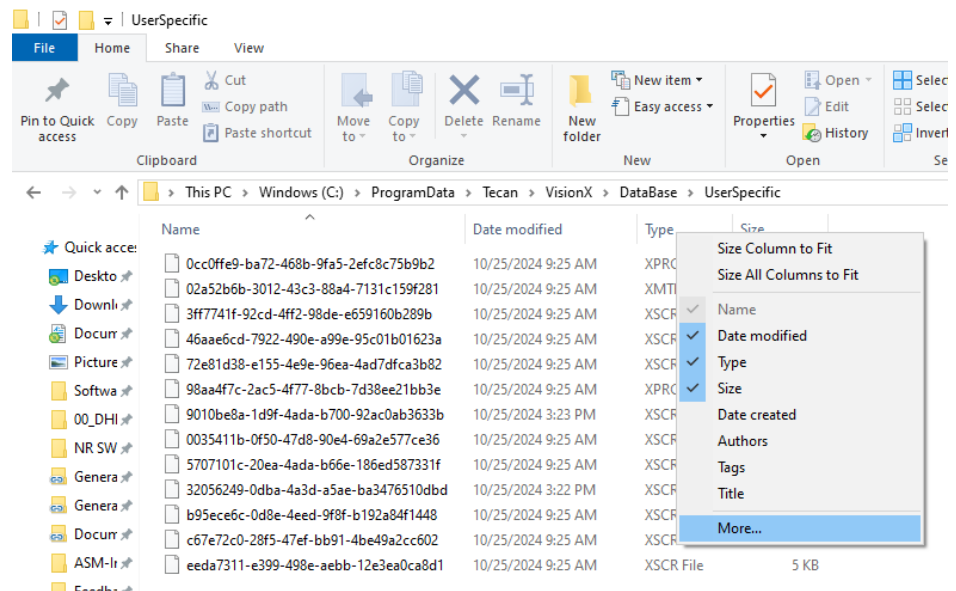
1. Create a Script with the command you want to add.
2. Parameterize and test the command.
3. Navigate to folder C:\ProgramData\Tecan\VisionX\DataBase\UserSpecific.
4. Open the script in a text editor like Notepad or Notepad++
5. Search for "<ScriptGroup>"
6. Copy the text from "<ScriptGroup>" to "</ScriptGroup>"
7. Add a new line with "B;" to your .gwl file and append the copied text to this line.
 Remove all line breaks which were added.
 The complete text block must be part of the same line.

To find your script, add a column showing the vControl names:

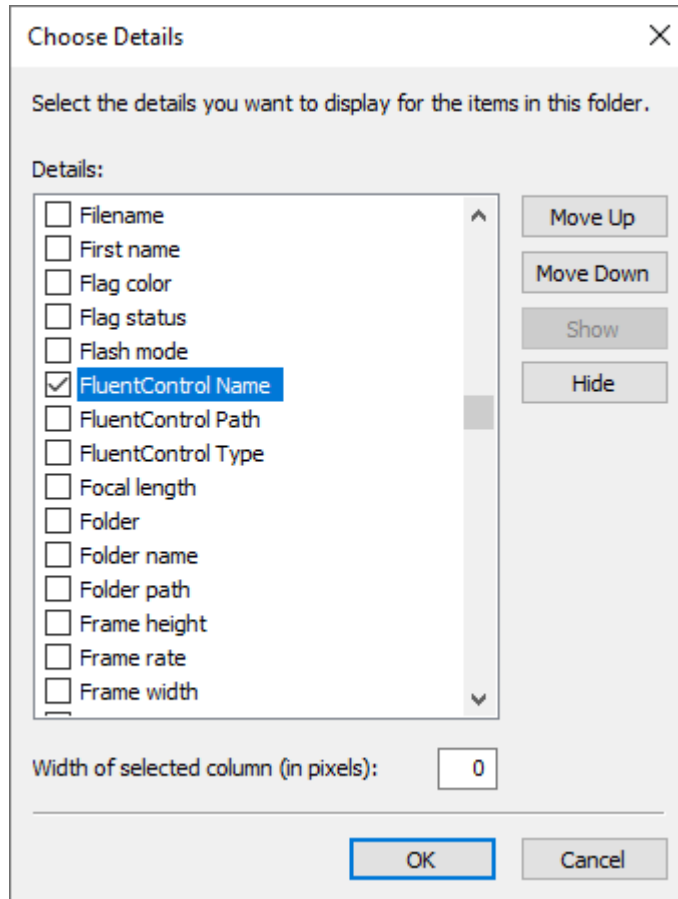
1. Open C:\ProgramData\Tecan\VisionX\DataBase\UserSpecific in the Windows Explorer.



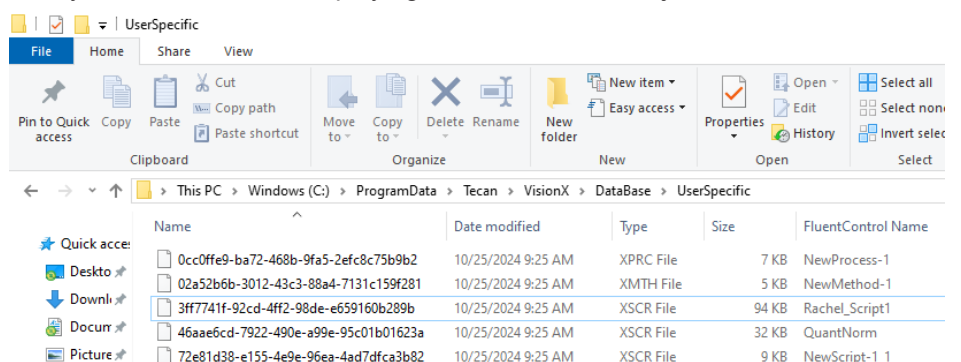
2. Right-click on any column header and select "More...".



3. Navigate to “FluentControl Name” and check the checkbox. Press OK.



4. Now you have a column displaying the names of the objects:



11.11.5 Execute Worklist

The **Execute Worklist** command processes the output of all **Load Worklist** commands which have been executed either since start of the script or since the execution of a last, previous **Execute Worklist** command, in the chronological order they were loaded.



vControl will at runtime create a temporary script from the worklist and execute that temporary script. This allows you to see exactly which steps have already been executed if you pause and resume the pipetting or are using Method recovery.

For very long worklists, displaying the temporary script may cause significant delays. You can avoid those delays when executing very long worklists by disabling the option **display sub-routine execution during runs** in **Configure System, General Settings, Application Settings**.



Fig. 149: Application settings

11.12 Sample Tracking

The **Sample Tracking** functions are controlled by special script commands.

Sample Tracking is a software add-on for vControl. It allows you to record sample and labware identities, track samples and labware through pipetting and robotic steps and to store process-related information, raw data and reduced data. The recorded data is stored centrally in an SQL database.

For more information about Sample Tracking, refer to the Sample Tracking Manual in the Documentation folder on the vControl installation medium.



Sample Tracking is not designed to work with duplicate barcodes. If duplicate barcodes are part of the workflow, make sure they are renamed to be unique before the Labware is registered in Sample Tracking. For example, suffixes or index numbers may be added using the "GetAttribute()" and "SetAttribute()" functions.



The following special characters must not be used in barcodes: ! % & ~ () = | as well as carriage return and line feed.

Prerequisites:

- ✓ Make sure source/destination labware has barcodes.
- ✓ Register source labware that contains liquid that you want to track.
- ✓ Do not register destination labware!

1. Run Step 3 Install Sample Tracking of the vControl Installer and install **Sample Tracking Embedded**.
2. On the Settings menu, click **Configure System** and navigate to **Drivers > Sample Tracking Driver**.
Sample Tracking driver in the vControl setting page is enabled.
3. To determine the format for the reports, use the **Generate Report** command in your script.

11.12.1 Sample Tracking, Commands, Overview

Tab. 15: Overview of Sample Tracking commands

Command	Cross reference to the Sample Tracking manual
Register Labware	RegLabware - This command is used to register labware and cavities.
Generate Report	PrintRep - This command is used to automatically generate printed reports.
Set Values	SetValues - The SetValue command provides the setting of values of identified tubes and wells.
Get Value	GetValue - The GetValue command is used to retrieve values of identified tubes or wells in Sample Tracking to script variables.

11.12.2 Generate Report

The **Generate Report** command is used to generate a sample tracking report as an execution step in a script. Optionally a printed report and a CSV file (e.g. for machine readable export) can be created. The output path for the CSV files can be adjusted under Configure System > Sample Tracking Driver > Report directory. Multiple labware selection is supported and for each selected labware separate reports are being created. PDF reports can be found here:

C:\ProgramData\Tecan\Sample Tracking\Reports



Sample Tracking reports will:

- Flag the sample as “Nothing was pipetted” if the error handling option “Ignore and continue” is chosen after an Aspiration Supervision Error.
 - Flag the sample as “Not enough liquid” if the error handling option “Dispense back and deactivate tip” is chosen after an Aspiration Supervision Error.
- Sample Tracking will show a warning for any pipetting steps that were interrupted by an Active Stop.*

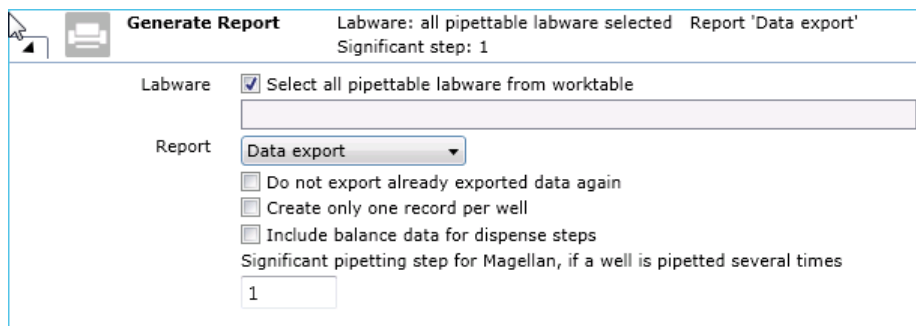


Fig. 150: Sample Tracking: Generate Report

Parameters

Labware

This panel shows the labware for which vControl will create output files. A separate report file is created for each labware. Specify a single or multiple labware for which a report shall be generated.

Add labware to the list by first clicking in the **Labware** box and then clicking on the desired labware on the worktable. To remove an entry, select the line in the list and press the **Delete** key (or click the labware on the worktable a second time).

To create reports for all labware on the worktable, select the **Select all pipettable labware from worktable** check box. In this case, the list of labware you have specified is ignored. The description then shows that all labware is selected.



A tube runner will be treated like a regular labware element: the tubes are treated like wells and the tube runner is treated like the labware element.

Do not export already exported data again check box

Do not export already exported data later again. If you use the **Generate Report** command more than once in the same script, select this check box if you don't want to export pipetting data again which has already been exported. This setting deletes the internal pipetting records for all prior pipetting operations, not only for the labware specified in the Generate Report command.

Create only one record per well check box

vControl normally outputs one record for each **Aspirate** or **Dispense** command for each well. If you select this check box, vControl only writes one record for each well. The record will then contain information on the accumulated errors (bitwise addition of all error codes) and the accumulated total pipetting volume (the sum of dispense volumes—i.e., positive values—and aspirate volumes—i.e., negative values).



Sample Tracking does not track the detected volume. In the 'Data Export' report, the column 'DetectVol' is therefore empty.

Include balance data for dispense steps check box

The **Liquid Arrival Check (LAC)** feature verifies that the correct liquid volumes have been dispensed by weighing the labware before and after pipetting. The labware must be put onto a special balance (pooling balance) for the dispense operation. Select this check box if you want to include the weight measurements in the report file. In this case, the report file will include four extra columns if a **Liquid Arrival Check** was carried out when dispensing.

Significant pipetting step for Magellan, if a well is pipetted several times box

Specify the relevant pipetting step for Magellan.

If you plan to use a Tecan Microplate Reader (photometer) in your script, specify which pipetting step is relevant to the Magellan data reduction software if the same well will be pipetted several times. For example, if you pipette the sample and then pipette a reagent, Magellan needs to know which step pipettes the sample. Information on the significant pipetting step will be included in the labware report file which the **Generate Report** command creates for the Magellan software.



If a report shall be generated that references data stored in external files, such as a plate map CSV files, the required files must be stored locally and not on a network location.

11.12.3 Get Value

The **Get Value** command is used to download values of registered tubes or wells from the Sample Tracking server and to store these values locally in script variables of vControl. This command is only used if the labware, such as a microplate, is already known to the Sample Tracking server. Labware is identified by either labware barcode and position in labware or the tube/well barcode. The labeling of all items reflects the function it represents and varies from the titles in this description.

Fig. 151: Get Value

Identify: In this section the user can provide the identification information for a labware.

Labware barcode/position option: In this section the user can provide the identification information for a labware.

- **Labware barcode:** Expression to set a value for the labware barcode.
- **Position in Labware:** Expression to set a value for the container position in labware.

Tube/well barcode: Expression to set a value for the tube/well barcode for the container.

- **Sample Alias:** Optional value as an expression to set a value for the sample alias.

Get value into variable

Value to retrieve: ComboBox to retrieve a value to be stored in the given variable.

Variable: Expression to select a variable and a declare button.

11.12.4 Register Labware

The **Register Labware** command is used to register labware, cavities and the associated samples in the cavities. The labware selected for this command has to be accessible during the execution of this command. The labeling of all items reflects the function it represents and varies from the titles in this description.

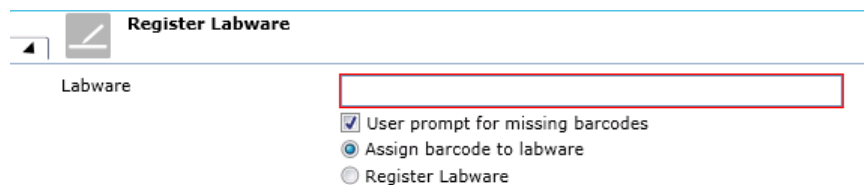


Fig. 152: Register Labware

Labware: Expression to select the available labware on the worktable.

AssignBarcode: To set if a single barcode is to be assigned for the whole labware if it has no barcode.

Register Labware: To generate barcodes for the labware and for all containers within if they have no barcode.



Concerning the labware selection: A tube runner will be selected as a single unit: the tubes are treated like wells and the tube runner is treated a piece of labware. It is not possible to select a single tube on its own. Clicking on a tube will cause its tube runner to be selected.

11.12.5 Set Values

The SetValue command is used to modify values on labware entries on the Sample Tracking Server. The command can thus only be used if the labware to be edited (e.g., microplate) has been previously registered on the Sample Tracking server. The SetValue command provides the setting of values of identified tubes and wells. Labware is identified by either labware barcode and position in labware or the tube/well barcode. The labeling of all items reflects the function it represents and varies from the titles in this description.

The **Set Values** command is used to modify values on labware entries on the Sample Tracking Server.

Fig. 153: Set values

Identify: In this section the user can provide the identification information for a labware.

Labware barcode / position

- **Labware barcode:** Expression to set a value for the labware barcode.
- **Position in Labware:** Expression to set a value for the container position in labware.

Tube / well barcode: Expression to set a value for the tube / well barcode for the container.

- **Sample Alias:** Optional value as an expression to set a value for the sample alias.

Values to set: In this section, the user can set further values regarding the liquid found in the tube / wells.

- **Tube barcode:** Expression to set a value for the container barcode.
- **Volume:** Expression to set a value for the container volume.
- **Sample Concentration:** Expressions to set a value for the sample concentration.
- **Sample Concentration:** Expressions to set values for the sample concentration unit.

User defined value: Expression to set a user defined value.



The Set Values command can only be used if the labware to be edited (e.g., microplate) has been previously registered on the Sample Tracking server with the Register Labware command.

11.13 Introspect

Introspect is a cloud-based service that allows for real-time monitoring of instruments.

Fig. 154: Introspect

You can use the **Send Notification** script command to push a message to the paired Introspect smart phone application. The Introspect mobile app can be downloaded from the app stores for iOS and Android.

11.14 Worktable

11.14.1 Set Available Tips (Worktable) - Direct Command

Use the **Set Available Tips** command to set the next available position to fetch disposable tips from.

Then select the next available DiTi position in the labware. Mark the positions of the next available DiTi in the plan view of the labware layout. Now, vControl will update the next available DiTi position for all labware items matching the labware name defined by the user.

Parameter

Labware: In the **Labware** box, either type your expression or click the labware displayed in the **Worktable** editor window (*Worktable click*).

11.14.2 On Empty Tray

Use the **On Empty Tray** command to define a subroutine handling for the automatic replacement of empty DiTi trays of a specified type on the worktable.

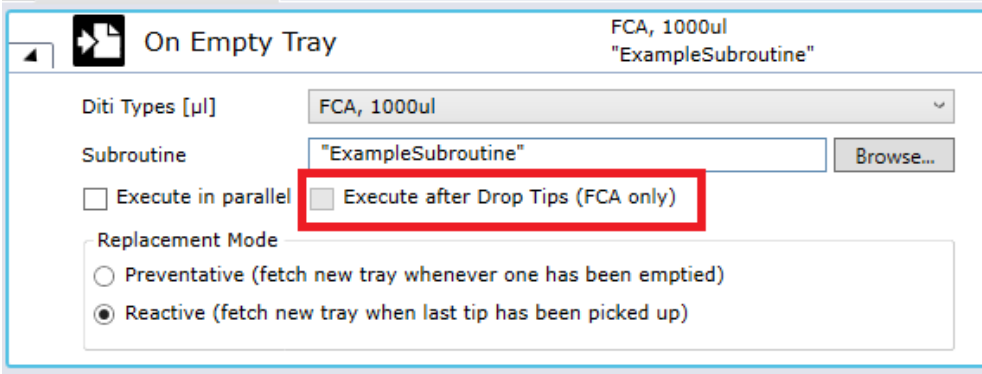


Fig. 155: On Empty Tray command

The **Execute after Drop Tips (FCA only)** option is intended for systems with only one FCA arm and no RGA when working in preventive mode. Unchecked the system replaces an empty tip box as soon as the box has been emptied, which usually happens during or right after a Get Tips command. However, this is not possible with only a FCA on the system, as FCA channel grippers can't be picked up while tips are mounted. Therefore, when the option is checked, the box will only be replaced after the tips have been dropped.



Make sure, there are always at least two tip boxes of the same tip type available on an accessible position on the worktable to ensure enough tips can be mounted for a given pipetting sequence.



The option **Execute in parallel** should only be chosen if you use the RGA to remove the empty tray and if your worktable layout permits that the RGA removes or replaces the empty tray while the FCA is pipetting.

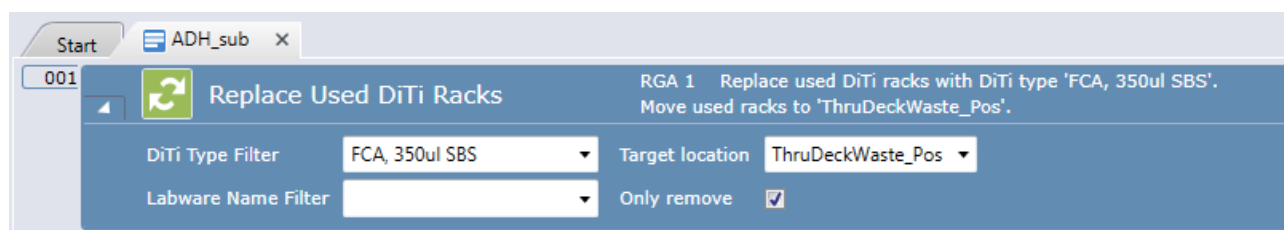


Fig. 156: Replace Used DiTi Racks command

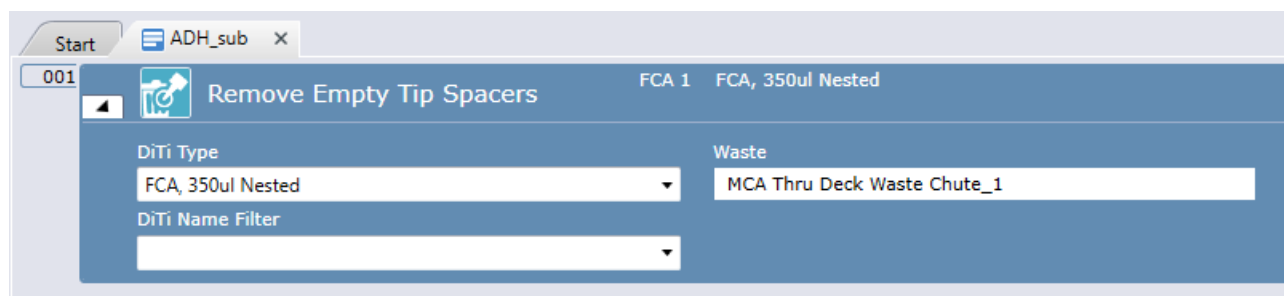


Fig. 157: Remove Empty Tip Spacers command

Typically, the subroutine that you link to will contain an RGA **Replace used DiTi Racks** command OR and FCA **Remove Empty Tip Spacers** command.

11.14.3 Import Sample IDs from File

Loads labware barcodes from a TXT or CSV file.

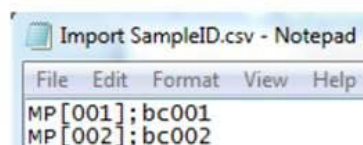
Path and filename of sample ID file box to specify a path and filename of the sample ID file box.

Prerequisites:

- The sample ID file must be a TXT or CSV file containing one sample ID per line.
- Each line consists of the labware name and the sample ID, separated by ',' or ';'.

1. Click **Browse**.
2. Specify the path and file name of the file containing the sample IDs to import.

LabwareLabel;LabwareBarcode



LabwareLabel;WellBarcode;WellIndex (where the well index is to be specified 1-based)

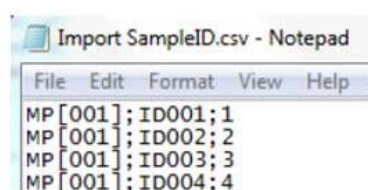


Fig. 158: Import sample IDs

11.14.4 Add Labware

Use the **Add Labware** command to programmatically populate the worktable with labware. You can then choose name, type, location and rotation of the labware at runtime.

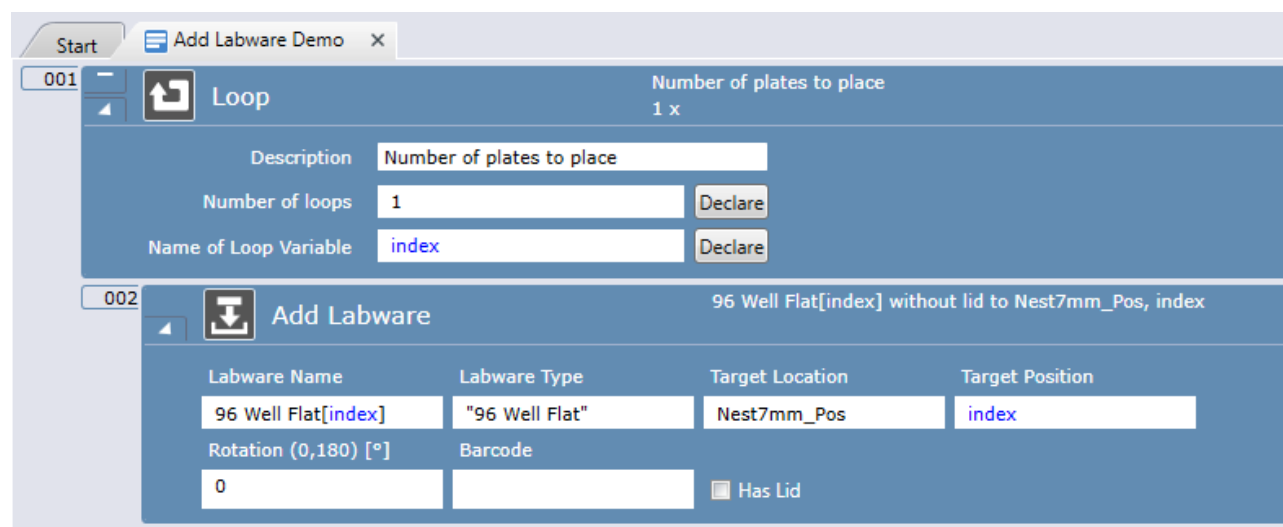


Fig. 159: Add Labware command

It is possible to select **Labware Type** and **Target Location** by clicking on the worktable or also by typing directly into the fields or by using variables.

11.14.5 Remove Labware

Use the **Remove Labware** command to programmatically remove labware from the worktable.



Fig. 160: Remove Labware command

11.14.6 Set Location

Use the **Set Location** command to programmatically set the location and rotation of a labware on the worktable. The set location command will not trigger any robotic action. You should use this command when the position of a labware changed after manual intervention of an operator or if the plate was moved using RGA single vectors or a 3rd party plate transport device.

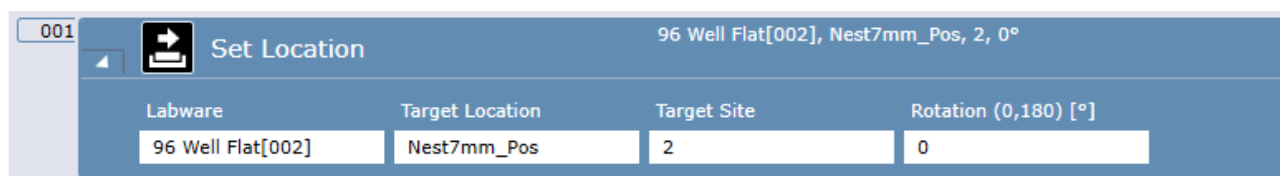


Fig. 161: Set Location command

12 Functions and Operators

For the examples below, the following variables were declared:

Name	Type	Value	Scope
DblVar1	Floating Point	12.34	Script
IntVar1	Integer	23	Script
StrVar1	String	StartValue	Script
ArrayInt1[]	Integer	0	Script

12.1 Operators Supported by Expression Solver

The following table lists all operators supported by the Expression Solver in order of precedence, from highest to lowest precedence:

Operator Symbols	Description	Example	Result
()	Parenthesis to group by evaluation order	3*(1+2)	9
^	power of	2^7 2^IntVar1	128 8388608
*	multiplication	DblVar1*8	98.72
/	division	230/IntVar1	10
DIV	modulo division (i.e. integer division)	IntVar1 Div 8	2
MOD	modulo	IntVar1 Mod 8	7
&	logical (bitwise) AND	0101 & 0011 3 & 5 3 & StrVar1	1 1 ERROR
+	addition, string concatenation	IntVar1 + DblVar1 "This value " + IntVar1 + StrVar1	35.34 This value 23StartValue
-	subtraction, unary minus (numeric negation)	IntVar1 – DblVar1 20 – IntVar1 -0.57 -DblVar1	10.66 -3 -0.57 -12.34
	logical (bitwise) OR	0101 0011 3 5 3 StrVar1	111 7 ERROR
XOR	logical (bitwise) XOR	0101 XOR 0011 3 XOR 5 3 XOR StrVar1	110 6 ERROR

Operator Symbols	Description	Example	Result
:=	Variable Name := Value Assigns Value to the Variable Returns Value as the result	Set StrVar1 to IntVar1:=50	StrVar1 = 50 IntVar1 = 50
=	Equality	IntVar1=DbiVar1	0
<>	Inequality	IntVar1<>DbiVar1	1
<	less than	IntVar1<DbiVar1	0
<=	less than or equal	IntVar1<=23	0
>	greater than	IntVar1>=DbiVar1	1
>=	greater than or equal	IntVar1>DbiVar1	1
NOT	logical negation	NOT 1 NOT 0 IntVar1=23 AND NOT IntVar1=0 NOT IntVar1=23 AND IntVar1=0 NOT (IntVar1=23 AND IntVar1=0)	0 1 1 0 1
AND	conditional AND	IntVar1 =23 AND IntVar1=0 IntVar1 =23 AND IntVar1<>0 1 AND 1 1 AND 0 0 AND 0	0 1 1 0 0 0
OR	conditional OR	IntVar1=23 OR IntVar1=0 IntVar1=23 OR IntVar1<>0 IntVar1<23 OR IntVar1<>0 1 OR 1 1 OR 0 0 OR 0	1 1 0 1 1 0

The relational and comparison operators (=, <>, <, <=, >, >=) allow the use of numeric as well as string operands. Both operands in a comparison are required to be of the same type. The bitwise operators require both operands to be of integral type. The use of parentheses to influence the order/precedence of evaluation is allowed.

12.2 Functions

The following table lists the basic functions supported by the Expression Solver:

Function	Parameters	Description	Example	Result
abs(x)	x: numerical value	Returns the absolute value of x.	abs(DbIVar1-IntVar1)	10.66
AdjustAccuracy()	Many	See AdjustAccuracy Function.xls	Located in the documents directory on the installation CD	
AdjustRangeBy-Channel()	Many	See AdjustAccuracy Function.xls	Located in the documents directory on the Installation CD	
AvailableVolume(labwareName, wellIndex, compartmentIndex, submergeDepth)	A float value in μ l	The driver allows to access the usable volume (with submerge depth and up to z-max) for a given well through a function—e.g., AvailableVolume("Microplate[001].Well[1]", 1, 1, 1.5)) . labwareName is the label of the labware, wellIndex the index of the well (1-based), compartmentIndex the index of the compartment within the well (usually 1) and submergeDepth the distance the tip goes into the liquid—i.e., a positive number means the tip is closer to the worktable).		
avg(x1,x2,...xn)	x1,x2,...xn: list of values	Returns the average of the specified values.	avg(DbIVar1+IntVar1,IntVar1,IntVar1)	27.1133333333333

Function	Parameters	Description	Example	Result
binary(x,[L])	x: integral value L: minimum length of returned string (padding with leading zeros if necessary; optional)	Returns the binary (string) representation of an integral value.	binary(IntVar1) binary(IntVar1,8)	10111 00010111
CheckIfPhase-HeightIsInExpectedRange(LiquidLevelBeforePhaseDetection, PhaseHeight, ConfidenceLevel)	LiquidLevelBeforePhaseDetection: The liquid level before starting the phase detection [mm] PhaseHeight: the height of the phase [mm] ConfidenceLevel: allowed values are 90%, 95% and 99%	Checks if a detected phase height is in an expected range of a blood sample.	CheckIfPhase-HeightIsInExpectedRange("120", "80", 95%)	-1: phase is below expected range 0: phase is in expected range 1: phase is above expected range
ceil(x)	x: numerical value	Rounds the given value to the nearest higher integer value.	ceil(3.01) ceil(3.995) ceil(3)	4 4 3
concat(s1,s2,...sn)	s1,s2,...sn: list of strings	Concatenates the specified strings.	concat("This value ", IntVar1, StrVar1)	This value 23StartValue
count(var)	var: array variable	Returns the number of elements present in the specified array (not the upper bound).	count(ArrayInt1[])	0
CurrentUser()	none	This function returns the name of the currently logged in user.	CurrentUser() Dependent on logged in user	No Login Required Admin Etc.
date()	none	Returns the current date in YY-MM-DD.	Date()	15-07-27
decimal(x)	x: binary string	Returns the integral value of a binary string.	decimal(11011) decimal(2)	27 Error

Function	Parameters	Description	Example	Result
DetectedVolume(<i>arm</i> , <i>tip</i>)	<i>arm</i> : Arm Name <i>tip</i> : Tip Index	If liquid level detection is used, the FCA pipetting commands automatically store the detected volumes for each tip in µl. The value is updated by each subsequent FCA pipetting command which uses liquid level detection. The value -1 is reported if the tip was used with liquid level detection disabled or was not part of the pipetting command. This also applies to the Detect Liquid command.	DetectedVolume("FCA 1", 1)	0 (or most recent volume)
ElapsedTime(<i>x</i>)	<i>x</i> : integer of started timer	Function returning the number of seconds having elapsed since the specified timer was last (re)started in the current script.	ElapsedTime(1) ElapsedTime(StrVar1)	305.4080805 (for example) ERROR
FileExist(<i>str</i>)	<i>str</i> : full path and file name	Check if a file exists and return "True" or "False".	FileExist(StrVar1) FileExist("C:\Program Files (x86)\Tecan\Control\ControlSW.exe")	=1 (meaning True) <>1 (meaning False)
FileLineCount	File Path	Use to count the number of lines in a text file. Returns an integer with the number of lines before the end of file.	FileLineCount(filePath)	5
FileExcelRowCount	file path, (optional) column, (optional) start row	Use to count the number of rows in an excel file until the first blank row. Returns an integer. Column and startRow are optional parameters.	FileExcelRowCount("C:\excelFile.xlsx")	150
floor(<i>x</i>)	<i>x</i> : numerical value	Rounds the given value to the nearest lower integer value.	floor(3.01) floor(3.995) floor(3)	3 3 3
FormatDateTime(double date-Time, string format = YYYY-MM-DD HH:MM:SS)	Parameter format is optional.	If format is not provided, the function takes the yyyy-MM-dd HH:mm:ss as the default format. Otherwise the function takes the given format and converts the DateTime/ Date from a long number to a string. Example formats	FormatDateTime("636841440000000000", "yyyy-MM-dd")	2019-01-27

Function	Parameters	Description	Example	Result
Format(double number, string format)	Both parameters are required.	Takes a number, and uses a format to format the given number. Example formats	Format(1252643,"e2") Format(123.456,"f2")	1.25e+006 123.46
GetAttribute	labware-Name, attribute	Gets a labware attribute like barcode, well concentration or custom attribute. For getting the labware barcode attribute name "Barcode" or "ID" can be used. For well barcodes use "Well[<well index>].Barcode" where <well index> is the 1-based well index. Likewise you can query the well concentration via "Well[<well index>].Concentration".	GetAttribute("FCA_1000ul Filtered[001]", "ID") GetAttribute("Source[001]", "Well[1].Barcode") GetAttribute("Source[001]", "Well[96].Concentration")	Current value of the attribute. Examples: "MyBC 1" 0.4
GetExpectedPhaseHeight(labwareName, well, compartment, liquidVolume)	labware-Name: name of the labware. well: 1-based well index. compartment: 1-based compartment index. liquidVolume: volume in [µL]	Gets a height in mm that represents the expected height of a phase in a blood sample.	GetExpectedPhaseHeight("G15ml_Falcon[016]", "1", "1", "6000")	34.45 [mm] (or other value for expected phase height [mm])
GetCoverSiteName & GetCoverSiteIndex	labware name	Returns the location and site names of given plate for where a lid or a stacked labware would rest.	GetCoverSiteName (96 Well Flat[001]) & GetCoverSiteIndex(96 Well Flat[002])	GetCoverSiteName = 96_Lid_Site & GetCoverSiteIndex = 3
GetExpectedStepStartTime	script name	Returns the expected start time for the iteration of the process step script. This can be used in combination with the Send Notification command to push the information to the Introspect to e.g. show the user when a specific step is finished.	GetExpectedStepStartTime("MyScript1")	21-09-17 15:43:55
GetLabwareByLocation	location name, position	Returns the labware name from a location at a specified position.	GetLabwareByLocation("Incub6_Pos", 4)	384 Well[001]

Function	Parameters	Description	Example	Result
GetLabwareLabel	barcode	Use to fetch plates from locations when only the barcode is known. For example, can be used with a carousel after an inventory scan to fetch a plate to use with a worklist.	GetLabwareLabel("BC12345")	MTP96[001]
GetLocation & Get-Site	labware name	Returns the location name and site number of a labware. Can be helpful to dynamically alter targets of RGA Execute Vector commands.	GetLocation(Labware- Name) & Get- Site(Labware- Name)	GetLocation = "Incub6_Pos" & GetSite = 3
GetLocationName-OfCarrier(<Carrier-Label>,<SiteIndex as shown in carrier editor>)	CarrierLabel, SiteIndex (optional, default value = 1)	Returns the location name of a site on a carrier that can be used, for example, in transfer-labware or add-labware commands	GetLocationNameOfCarrier("Nest 61mm[007]") GetLocationName-OfCarrier("6 Landscape 61mm Nest[001]",7)	"Nest61mm_Pos" WasteSlide_Site
GetNextIdleTime-ForResource	resource-Name minimumDurationOfIdle-Time (in minutes)	Returns the next idle time which fits the given minimum duration for the given resource. If no idle time is found the end of the run will be returned.	GetNextIdleTime-ForResource("My-Washer", 5)	21-09-17 15:56:14
GetNumberOfTipsInTipBox	labware name	Returns the number of available tips in a box.	GetNumberOfTipsInTip-Box("FCA_ 1000ul Filtered[001]")	96
GetTimeSinceSuccessfulExecution	method name	Returns the time since a method was successfully executed. Useful if analysis on when a system care method's last run time may impact a given run.	Get- TimeSince-SuccessfulExecution(Method- Name)	2.25
GetTipBoxStatus	labware name	Returns the tip status of a tip box, with 0 and 1 indicating which positions of tips in a box are in use or available.	GetTipBoxStatus("FCA_ 10ul SBS[001]")	"0101010101111110000011001"
GetLoading IDLED Status & GetLoading IDSensorStatus	grid number	Returns the LED status and sensor status of the Loading ID 5-grid module. Useful to dynamically interact with operators at run-time.	GetLoading ID 5-grid LEDStatus(Grid) & GetLoading ID 5-grid SensorStatus(Grid)	GetLoading ID 5-grid LEDStatus = 1 & Loading ID 5-grid SensorStatus = 0

Function	Parameters	Description	Example	Result
GetLocationName-OfCarrier(Carrier-Label, [SiteIndex])	CarrierLabel name of the carrier on the worktable SiteIndex (optional, default value = location name of site1)	Returns the location name to be used in a transfer labware or add labware command.	GetLocationName-OfCarrier("7mm Nest[009]")	Nest7mm_Pos
GetPositionIndex-OfCarrier(Carrier-Label, [SiteIndex])	CarrierLabel name of the carrier on the worktable SiteIndex (optional, default value = location name of site1)	Returns the target position to be used in a transfer labware or add labware command. The position is counted over the full worktable whereas the SiteIndex is within one carrier.	GetPositionIndex-OfCarrier("7mm Nest[009]")	3
if(c,e1,e2)	c: condition e1: expression e2: expression	Evaluates the condition c. If c yields true, the value of e1 is returned, otherwise the value of e2.	if(IntVar1<DbiVar1,"True","False") if(IntVar1<DbiVar1 OR IntVar1>DbiVar1, 1+4,"False")	False 5
InStr(inputString, searchString, numberOfOccurrence, startPosition, ignoreCase = false)	numberOfOccurrence, startPosition, ignoreCase are optional parameters	If the search string is found in Input string, the function returns the index number (zero based) depending on the other parameters provided. Returns -1 if not found.	InStr("abc-cba","b",2)	4 (start position of the 2nd occurrence of the string "b" is 4)
LabwareCount(labwareGroupName)		Returns the number of labware (i.e., tubes) that have a barcode assigned. Usually used to count the number of tubes scanned with Loading ID 5-grid to use this number as sampleCount for further processing	Labware-Count("G13x100mm")	18 (assuming that 18 13x100mm tubes were scanned)
LastErrorId()	none	This function returns the error message ID of the last command which reported an error during script runtime, example APP_xxx_yyy_zzz.	LastErrorId()	Depends on the error

Function	Parameters	Description	Example	Result
LastErrorMessage()	none	This function returns the error message of the last command which reported an error during script runtime.	LastErrorMessage()	Depends on the error
LastErrorSeverity()	none	This function returns the error severity of the last command which reported an error during script runtime as an integer.	LastErrorSeverity()	Depends on error
length(x)	x: any variable, string, constant or expression	Returns the total number of characters of the parameter.	length(StrVar1) length(IntVar1) length(DbVar1)	10 2 5
ln()	x: numerical value	Returns the natural (base-E) logarithm of the specified value.	ln(100) ln(IntVar1)	4.605170185 98809 3.135494215 92915
log()	x: numerical value	Returns the base-10 logarithm of the specified value.	log(100) log(IntVar1)	2 1.361727836 01759
max()	X1,X2,...Xn: list of values	Returns the maximum of the specified values.	max(DbVar1,IntVar1,17,20.6) max(DbVar1,IntVar1,StrVar1)	23 ERROR
min()	X1,X2,...Xn: list of values	Returns the minimum of the specified values.	min(DbVar1,IntVar1,17,20.6) min(DbVar1,IntVar1,StrVar1)	12.34 ERROR
ParseDateTime(string date-Time, string format = YYYY-MM-DD HH:MM:SS)	Parameter format is optional.	If format is not provided, the function takes the yyyy-MM-dd HH:mm:ss as the default format. Otherwise the function takes the given format and converts the DateTime/ Date to a long number.	ParseDateTime("Jan27-2019", "MMMdd-yyyy")	63684144000 0000000
pow(x,y)	x: the base y: the power	Raises the specified base x to the power y.	pow(IntVar1,3) pow(3,3)	12167 27
random()	none	Generates a random floating-point number between 0 and 1. Is seeded by the system clock.	random() round(random()*(IntVar1-1))+1	0 to 1, for example: 0.325044045 841808 Integers from 1 to 23

Function	Parameters	Description	Example	Result
round(x,[p])	x: numerical value p: number of fractional digits (optional)	Rounds a specified value to the specified number of fractional digits (0 if not specified).	round(0.49) round(0.5) round(round(0.49,1)) round(10,2) round(10.195,2)	0 1 1 10 10.2
ScriptIteration()	none	Returns the iteration of the current script.	ScriptIteration()	1
ScriptName()	none	Returns the name of the current script.	ScriptName()	NewScript-1
SerialNumber()	none	Returns the serial number of the connected instrument.	SerialNumber()	1901001234
SetAttribute	labware-Name, attribute, value	Sets a labware attribute like the barcode. For setting the barcode "Barcode" or "ID" can be used. For well barcodes use "Well[<well index>].Barcode" where <well index> is the 1-based well index. Likewise you can set the well concentration via "Well[<well index>].Concentration".	SetAttribute("FCA_1000ul Filtered[001]", "ID", "MyBC") SetAttribute("Source[001]", "Well[1].Barcode", "Well1") SetAttribute("Source[001]", "Well[96].Concentration", "0.4")	The assigned value
SetTipBoxStatus	labware name, status	Sets the tip status of a tip box.	SetTipBoxStatus("FCA_1000ul Filtered[001]", "010101110100011111111010101010101")	1 = successful, otherwise 0
sqrt(x)	x: numerical value	Returns the square root of the specified value.	sqrt(9) sqrt(lnVart1)	3 4.79583152331272
substring(s,start,[count])	s: string value start: start position count: optional length	Extracts a substring starting at position start from specified string; optionally limiting the length of the returned string.	substring(StrVar1,5,3) substring(StrVar1,2)	Val artValue
time()	none	Returns the current time in 24 hour format: HH:MM:SS.	time()	21:55:27
WorktableName()	none	Returns the name of the current worktable	WorktableName()	Base Worktable

12.3 Operands

An operand used in an expression can be any of the following:

- fixed numerical value, using the '.' character as decimal separator in case of a floating-point value
- string constant enclosed in double-quotes
- variable
- carrier, labware or location attribute

The use of array variables is permitted. Array elements are accessed using brackets to enclose the index; multi-dimensional arrays are supported. An array index itself may be an expression. To declare an array variable, enter the variable name directly in the Variables panel and add [] to the end of the name, for example arrayVar[] (see figure above). It is also possible to declare a multi-dimensional array by using a series of [] brackets: arrayVar[][][].

Variables in expressions are specified without scope identifier(s). The Expression Solver uses the scope of the current script to access the Variable Engine, which in turn tries to resolve variables within that scope first, working its way upwards in the scope hierarchy until the variable is found.

Carrier/Labware or location attributes are resolved using the current workspace. (At runtime, there is exactly one current workspace for all running scripts and/or processes.)

The labware index, site index and well number can be specified using a numeric expression.

12.4 Worktable Lookup Functions



All results in this table are dependent on labware selected and worktable layout.

This section documents the functions provided by the expression solver to lookup different attributes of labware and cavities:

Function	Parameters	Description	Example	Result
alphaNumeric(wellIndex, labwareName, [lengthOfNumericPart])	wellIndex: index of well within labware (1-based) labwareName: label of labware lengthOfNumericPart: optional length of numeric part in output	Returns the alphanumeric well identifier corresponding to the given well index, according to the orientation of the labware on the worktable.	alphaNumeric(19, "96 Well Flat[001]", 3) alphaNumeric(96, "96 Well Flat[001]")	C003 H12

Function	Parameters	Description	Example	Result
calculate-Height(labwareName, well, compartment, volume)	labwareName: LabwareName[Index] well: LabwareName[Index].Well[WellNo] compartment: index > 0 volume: volume in µl	height within cavity in mm (floating-point)	calculate-Height("96 Well Flat[001]", 1, 1, 100)	3.013584 72126666
calculateVolume(labwareName, well, compartment, height)	labwareName: LabwareName[Index] well: LabwareName[Index].Well[WellNo] compartment: index > 0 height: height within cavity in mm	volume at specified height in µl (floating-point)	calculateVolume("96 Well Flat[001]", 1, 1, 2.3)	76.32106 6528147
column(loopIteration, labwareName)	loopIteration: current value of loop counter (> 0) labwareName: LabwareName[Index]	Well index to apply to pipette into column corresponding to loop counter.	column(3, "96 Well Flat[001]")	16
GetAttribute(labwareName)	labwareName: LabwareName[Index] attributeName: name of attribute to read	Retrieve the value of an attribute from a labware. Also allows accessing attributes of a well within the labware.	GetAttribute("Microplate[001]", "Barcode") GetAttribute("Microplate[001]", "Well[23].Volume")	"Sample-017" 73.2783
GetLocation(labwareName)	labwareName: LabwareName[Index]	Determines the location (= location group name) where a labware is currently placed on the worktable.	GetLocation("Microplate[001]")	"Nest_61 mm"

Function	Parameters	Description	Example	Result
GetSite(labwareName)	labwareName: LabwareName[Index]	Determines the site (= site index) within the location group where a labware is currently placed on the worktable.	GetSite("Microplate[001]")	1
numeric(wellIdentifier, labwareName)	wellIdentifier: alphanumeric identifier of well labwareName: label of labware	Returns the numeric well index corresponding to the given well index, according to the orientation of the labware on the worktable.	numeric(A004, "96 Well Flat[001]") numeric(H12, "96 Well Flat[001]")	25 96
plate(labwareTypeName, loopIteration)	labwareTypeName: name of labware type being pipetted loopIteration: current value of loop counter (> 0)	The label of the labware instance to use for pipetting, corresponding to the loop counter. (string)	plate("96 Well Flat", 4)	"96 Well Flat[004]"
quadrant(labwareName, loopIteration, headRows, headColumns)	labwareName: LabwareName[Index] loopIteration: current value of loop counter (> 0) headRows: number of rows on pipetting head headColumns: number of columns on pipetting head	Well index to apply to pipette into plate quadrant corresponding to loop counter.	quadrant("384 Well [001]", 3, 8, 12)	16
row(loopIteration, labwareName)	loopIteration: current value of loop counter (> 0) labwareName: LabwareName[Index]	Well index to apply to pipette into row corresponding to loop counter.	row(3, "96 Well Flat[001]")	2

Function	Parameters	Description	Example	Result
SetAttribute(labwareName,attributeName)	labwareName: LabwareName[Index] attributeName: name of attribute to write value: New value to assign	Update the value of an attribute of a labware. Returns the new value having been assigned.	SetAttribute("Microplate[001]", "Barcode", "Sample-123") SetAttribute("Microplate[001]", "Well[23].Volume", "150")	"Sample-123" 150
SetLocation(labwareName,location,Site)	labwareName: LabwareName[Index] location: Location group name Site: Index of site within location group (1 ... n)	Moves a labware to a given location and site on the worktable.	SetLocation("Microplate[001]", "Hotel_9Pos", 5)	
tipAccess(labwareName,[well, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	number of tips allowed to access the well in parallel	tipAccess("96 Well Flat[001]", 71) tipAccess("100 ml_1", 1)	1 8
tipTouchDistance(labwareName,well,compartment,direction[, height])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: index of compartment within well (1-based) Direction: 0 = X, 1 = Y height: optional height within cavity in mm; if not specified, current Z-axis position of calling channel is used	Tip-touching distance (i.e. distance to rim of cavity) at given height in [mm] (floating-point)	tipTouchDistance("96 Well Flat[001]", 1, 1, 0)	3.3

Function	Parameters	Description	Example	Result
xPosition(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	X-Position of well (center) in mm	xPosition("96 Well Flat[001]", 1)	209.3
yPosition(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Y-Position of well (center) in mm	yPosition("96 Well Flat[001]", 71)	350.775
zBottom(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Bottom coordinate in mm	zBottom("96 Well Flat[001]", 1, 1)	11.2
zDispense(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Dispense coordinate in mm	zDispense("96 Well Flat[001]", 1, 1)	19.6
zMax(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Max coordinate in mm	zMax("96 Well Flat[001]", 1, 1)	11.7

Function	Parameters	Description	Example	Result
zStart(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Start coordinate in mm	zStart("96 Well Flat[001]", 1, 1)	21.6
zTravel(labwareName, well[, compartment])	labwareName: LabwareName[Index] well: index of well within labware (1-based) compartment: optional index of compartment within well (1-based)	Z-Travel coordinate in mm	zTravel("96 Well Flat[001]", 1, 1) zTravel("96 Well Flat[001]", 1)	29.6 29.6

Tab. 16: Syntax table

Syntax	Example
LabwareName[Index].Attribute	Microplate[001].Barcode
LabwareName[Index].Well[No].Attribute	Microplate[001].Well[1].Volume
LocationName.Attribute	DitiWaste.Barcode
LocationName[SiteIndex].Attribute	MP4Pos[1].Barcode

Other Variable Pipetting Loop functions that can be used outside of a Variable Pipetting Loop:

Function	Arguments	Description	Example	Result*
tipsNeeded(samples)	samples: number of samples to pipette	Tip mask required to pipette the remaining number of samples. Same as $2^{\text{samples}} - 1$	tipsNeeded(5)	31

Function	Arguments	Description	Example	Result*
tipUseCount(tipNo, tipCount, samples)	tipNo: index of tip to evaluate (1-based) tipCount: number of tips available samples: total number of samples to pipette	Number of dispense steps required for the specified tip to pipette the total number of samples.	tipUseCount(3,8,27) tipUseCount(3,8,26)	4 3
tipsUsed(samples)	tipMask: bit-mask of tips selected for pipetting	Tip mask required to pipette the remaining number of samples. In addition, the function reduces the number of remaining samples by the number of selected tips.	tipsUsed(5)	31

13 Data Handling

This chapter provides information about the handling of user data. Log files may become huge. Depending on company standards you may wish to backup log files in regular intervals. After backup delete the log file to start afresh.

13.1 Saving User Data

Describes the concepts backup and restore of vControl.

If vControl is already installed on your PC, we recommend you to back up your vControl data files, scripts and processes to exchangeable media or to a network or temporary directory (not within the vControl <install_path>) before running the installation program.

Saving these files is especially recommended in the following cases:

- Before uninstalling (removing) the vControl software.
- Before installing a newer version of vControl.
- Before modifying an existing vControl installation.

13.2 Backup

When a computer write to disk action is interrupted, for example by power loss, it may be possible that restoring a backup is necessary to make vControl operational again. If the SVN Repository is impacted by such data corruption restoring a backup may be necessary to maintain the integrity of the Audit Trail.

Data to backup:

- **Database Folder:** This contains all the files necessary to run assays. It includes the definitions of Methods, Scripts, Liquid Classes, Carriers, Labware, etc.
 - **SVN-Repository:** This contains the change history for all objects in the database. It enables the database restore points and “rolling back” changes to individual objects.
 - **SVN-Password:** This is the system password vControl uses to communicate with the SVN. This is not a protected password, nor is it related to User Management.
 - **Audit Trail Folder** (optional): This should be backed up if the change history of vControl’s electronic records is critical to maintain in the event of computer failure.
1. Shutdown vControl.
 2. Shutdown SVN Service.
 3. Backup Target: Create new Folder if necessary.
 4. Copy Folders (with all subfolders and files) to Backup Target Folder: C:\ProgramData\Tecan\VisionX**Database**, C:\ProgramData\Tecan**SVNRoot** and optional C:\ProgramData\Tecan\VisionX\AuditTrail.
 5. Create Registry Export of SVN-Password (and other vControl data):
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Tecan.
 6. Following Command Line will do this for you: regedit /E BackupTargetFolder/Registry.reg HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Tecan.
 7. Start SVN-Service again.

To restore a backup:

1. Shutdown vControl.
2. Shutdown SVN-Service.
3. Delete current Folders: C:\ProgramData\Tecan\VisionX**Database** and C:\ProgramData\Tecan**SVNRoot**.
4. Copy backed up folders back to original source.
5. If registry entries were changed: Import Registry File.
This is required when vControl was removed completely and SVN-Password is lost.
6. Start SVN-Service again.

13.3 Restoring

1. Install vControl in the same version the backup has been run on.
2. Close vControl in case it is running.
3. Terminate all vControl processes in case they are running. These are for example:
Mcs.exe
Tecan.Licensing.ClientModule.exe
4. Stop the following services:
Tecan vControl SVN
Tecan vControl Watchdog
Tecan User Management Server 1.1
Tecan.LoggingService
5. Copy the data you previously backed up to their origin folders. Make sure the access rights of the files and directories are restored as they were on the backed-up system.
6. Restart the services that were stopped before.
7. Start vControl.

13.4 Archiving Log and Audit Trail Data

If you are required to comply with the FDA regulations, make sure archived copies of all Audit Trail and log files are kept. The relevant files are stored in the following directories and their subdirectories:

- C:\ProgramData\Tecan\VisionX\AuditTrail
- C:\ProgramData\Tecan\LoggingServer\
- C:\ProgramData\Tecan\DataStore\IoT-Client\MAP.Services.Logging.Service\LogFile
- C:\ProgramData\Tecan\Tecan User Management\v1.1
- C:\ProgramData\Tecan\Sample Tracking\AuditTrail\log

These directories are protected against unauthorized access. The AuditArch tool allows you to copy or move the log files and audit trail data files to another directory of your choice.

14 Error Handling and Recovery

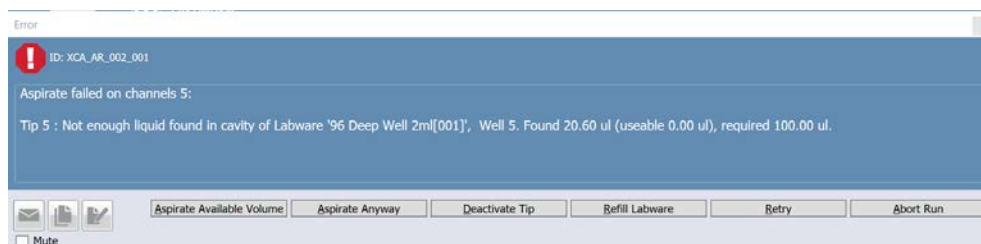
This chapter describes runtime error messages and the associated dialog boxes that can be issued by the software. It provides information on the cause of the error and instructs how to correct the error, where applicable.

14.1 Errors Requiring User Interaction (FCA)

This section describes pipetting error messages which require user interaction. Please see the description of the output file format of the Export Data command for information on the pipetting error codes which are reported. Errors are also reported in the application log files.

14.1.1 FCA Error: No Liquid / Not Enough Liquid

This error message is shown in case one or more channels detect an insufficient amount of liquid for aspiration.



The user has the following options to handle the error:

Aspirate Available Volume: The affected channels aspirate the available volume in the cavity (shown as “useable” in the error message).

Aspirate Anyway: The affected channels aspirate the requested volume.

Deactivate Tip: The affected channels skip the aspiration and are deactivated for the remainder of the current pipetting sequence, i.e. until tips are dropped.

Refill Labware: All channels are raised to their highest position, so as to give the user access to the labware. Another dialog is shown prompting the user to refill the labware, upon confirmation all channels retry the aspiration including the liquid level detection.

Retry: The affected channels retry the liquid level detection.

Abort Run: The entire run is aborted.

14.1.1.1 FCA error: DiTi(s) not dropped

Error received if one or more DiTis were not properly ejected.

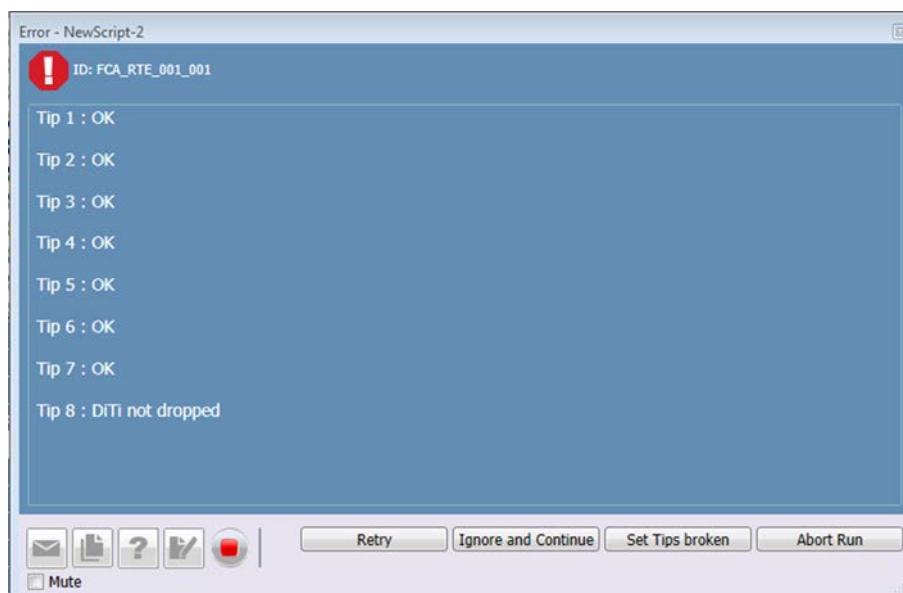


Fig. 162: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry:** The Drop Tips command executes again trying to eject the tips.
- **Ignore and Continue:** vControl continues the execution of the script regardless of whether the tips have been properly ejected.
- **Set Tips broken:** Channels with not dropped DiTis are marked as broken and are not used further in the pipetting sequence.
- **Abort Run:** The whole script is aborted.

14.1.1.2 FCA error: DiTi has been lost

Error received if one or more DiTis were lost during the execution of a script.

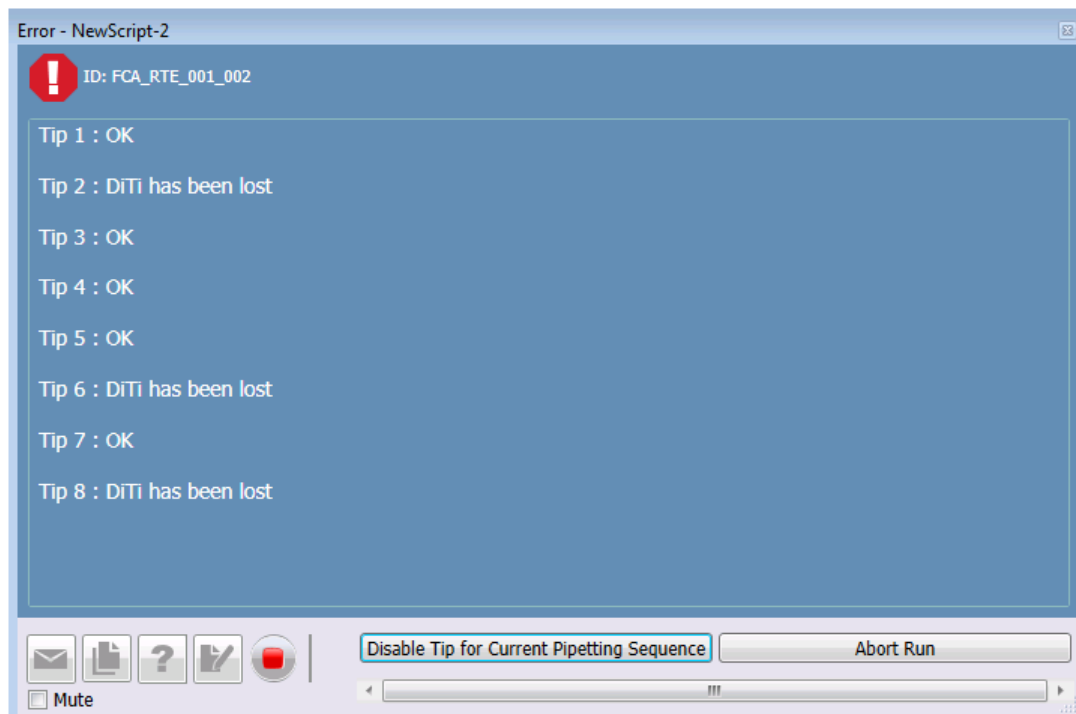


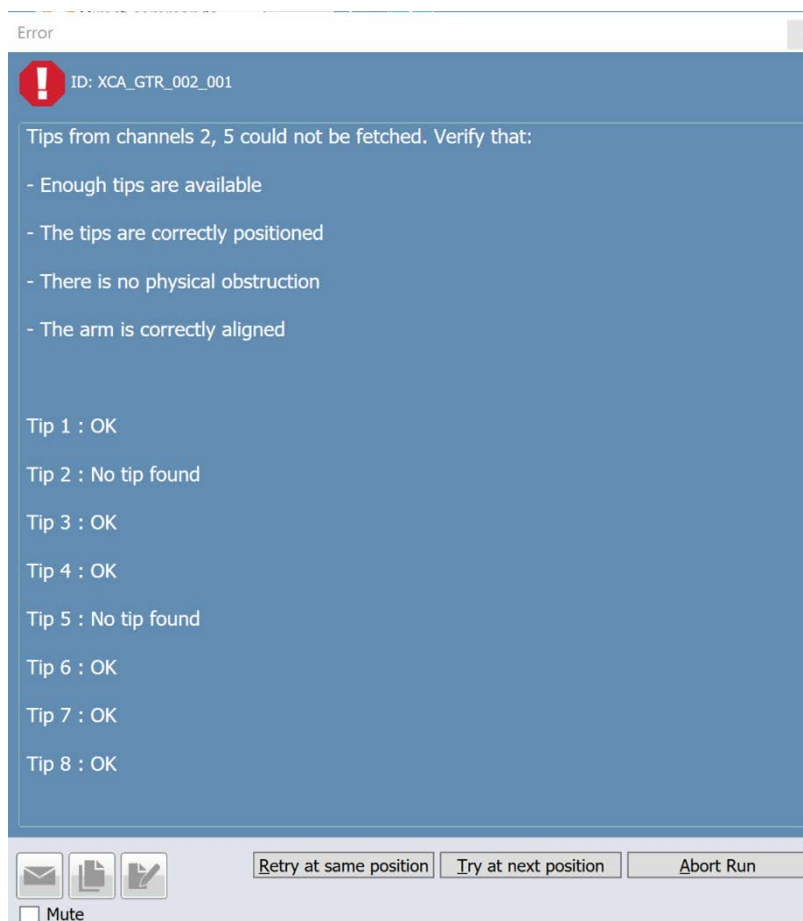
Fig. 163: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Disable Tip for Current Pipetting Sequence:** Channel(s) with lost DiTis are disabled for the current pipetting sequence and the execution of the script continues.
- **Abort Run:** The whole script is aborted.

14.1.1.3 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

Abort Run: The entire run is aborted.

14.1.2 FCA Error: Pressure Profile Error

If the monitoring of the pressure during a pipetting operation detects an error, the software reports a "Pressure Profile Error" for the affected channels and provides the following options to handle that error situation:

Continue, Pipette Nothing: The affected channels are deactivated for the remainder of the current pipetting sequence, i.e. until tips are dropped.

Continue, Dispense Back, Pipette Nothing: The affected channels dispense back the liquid to their respective source wells, then they are deactivated for the remainder of the current pipetting sequence.

Continue, Drop Tip, Pipette Nothing: The affected channels discard the disposable tips and are then deactivated for the remainder of the current pipetting sequence.

Retry Once with new Tip: The affected channels retry the aspiration once as follows:

1. Dispense back the liquid to their respective source wells.
2. Discard the currently mounted tips.
3. Fetch new tips of the same type.
4. Retry the entire aspiration.
5. If another pressure profile error occurs during retry, the error handling option Continue, Drop Tip, Pipette Nothing is executed automatically.

In any case, the affected channels perform an automatic channel self-test after dropping the tips.

14.1.2.1 FCA error: DiTi(s) not dropped

Error received if one or more DiTis were not properly ejected.

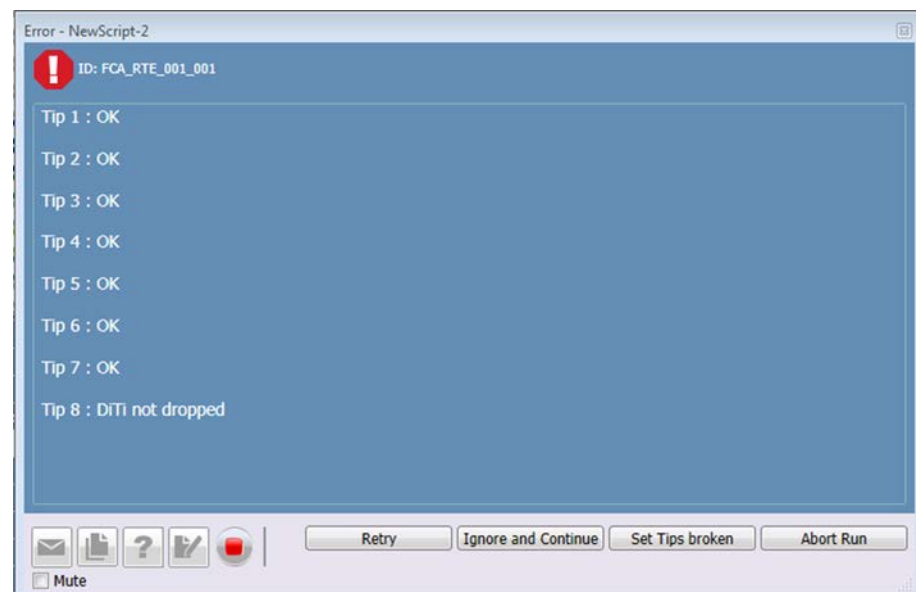


Fig. 164: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry:** The Drop Tips command executes again trying to eject the tips.
- **Ignore and Continue:** Continues the execution of the script regardless of whether the tips have been properly ejected.
- **Set Tips broken:** Channels with not dropped DiTis are marked as broken and are not used further in the pipetting sequence.
- **Abort Run:** The whole script is aborted.

14.1.2.2 FCA error: DiTi has been lost

Error received if one or more DiTis were lost during the execution of a script.

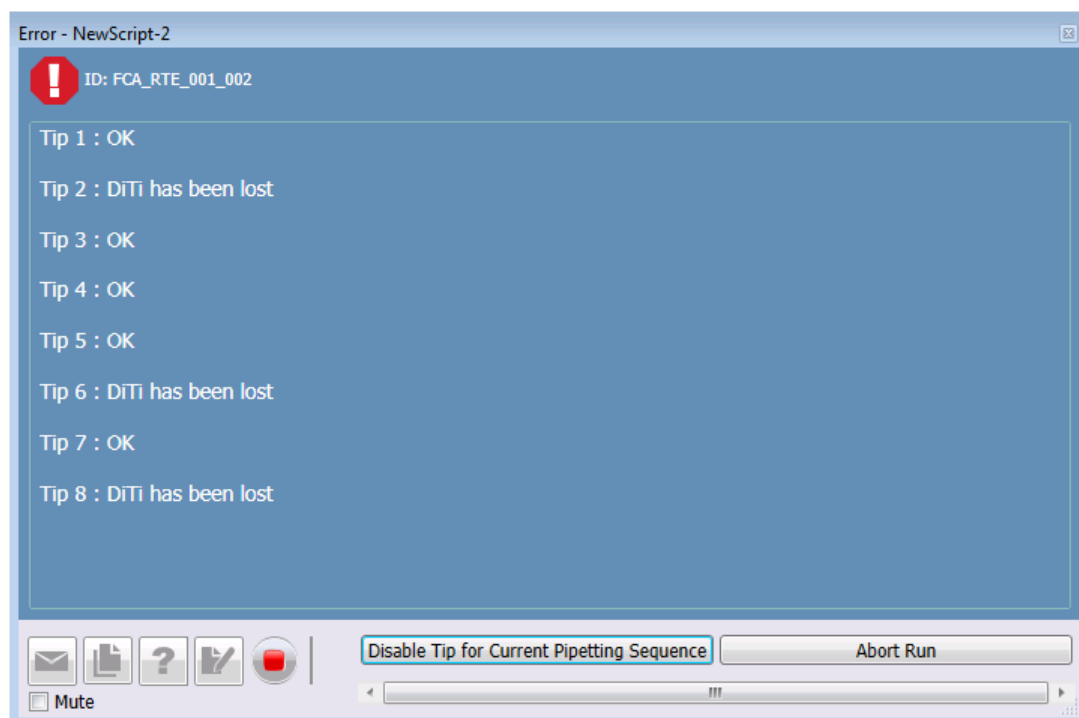


Fig. 165: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Disable Tip for Current Pipetting Sequence:** Channel(s) with lost DiTis are disabled for the current pipetting sequence and the execution of the script continues.
- **Abort Run:** The whole script is aborted.

14.1.2.3 FCA error: DiTi(s) not fetched

Error received if one or more DiTis were not properly fetched. The worktable is accessible for the user to fill the DiTi rack with new DiTis or to replace the whole DiTi rack.



When replacing the whole DiTi rack, put the new DiTi rack on exactly the same worktable position.

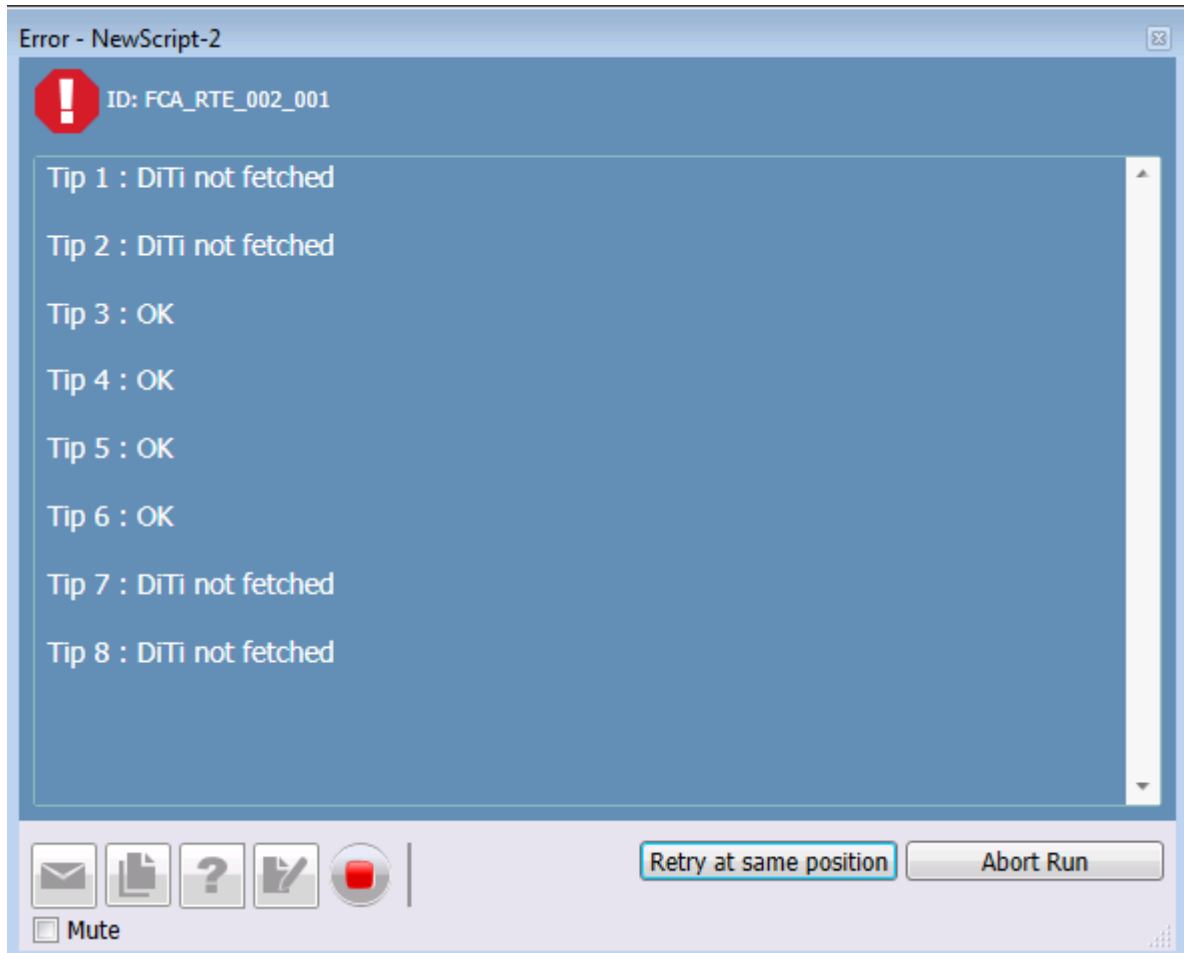


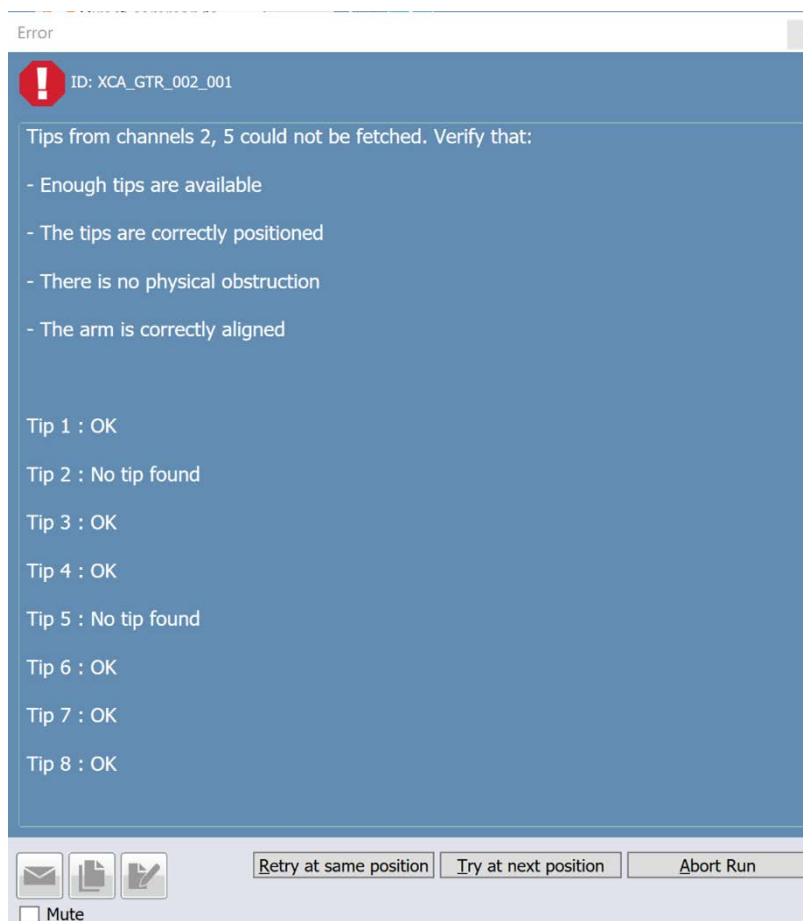
Fig. 166: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry at same position:** The FCA tries to get DiTis again from the same position as before.
- **Abort Run:** The whole script is aborted.

14.1.2.4 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

Abort Run: The entire run is aborted.

14.1.3 FCA Error: PMP Aspiration Error

If PMP is activated for a pipetting sequence and the evaluation model detects a potential aspiration error, the software reports that error to the user and provides the following options to handle the error situation:

Ignore and Continue: The affected channels ignore the error and continue pipetting. A warning message is shown in the System Trace View.

Dispense Back to Source and Deactivate Tip: The affected channels dispense back the liquid to their respective source wells, then they are deactivated for the remainder of the current pipetting sequence.

Dispense to Waste and Deactivate Tip: The affected channels dispense the liquid to a liquid waste, then they are deactivated for the remainder of the current pipetting sequence.

The software automatically selects a suitable liquid waste. A labware is designated as a liquid waste either by its type (LiquidWaste) or by a custom attribute **IsFcaLiquidWaste**.

Retry Once with new Tip: The affected channels retry the aspiration once as follows:

1. Dispense back the liquid to their respective source wells.
2. Discard the currently mounted tips.
3. Fetch new tips of the same type.
4. Retry the entire aspiration.

If another PMP aspiration error occurs during retry, the error handling option Continue, Drop Tip, Pipette Nothing is executed automatically.

14.1.3.1 FCA error: DiTi(s) not dropped

Error received if one or more DiTis were not properly ejected.

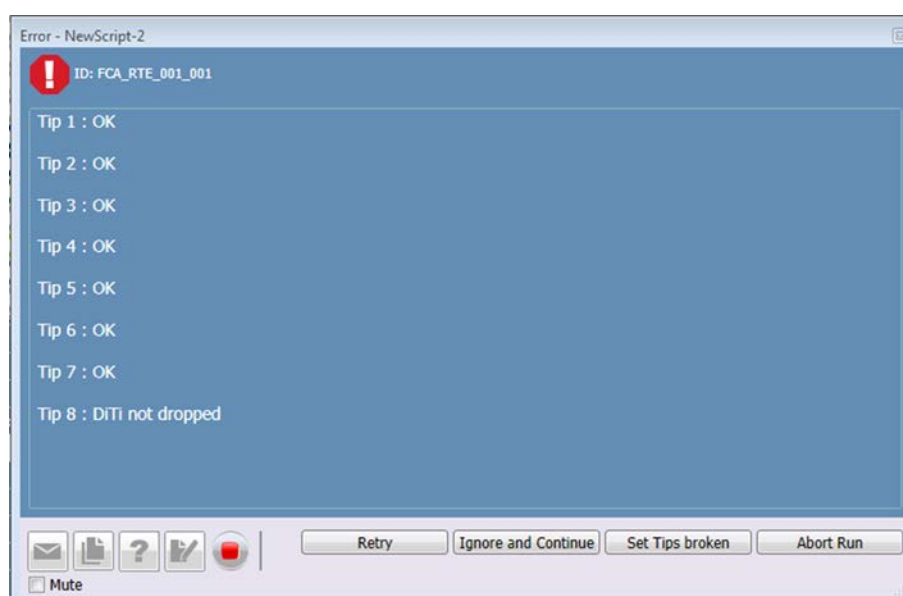


Fig. 167: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry:** The Drop Tips command executes again trying to eject the tips.
- **Ignore and Continue:** Continues the execution of the script regardless of whether the tips have been properly ejected.
- **Set Tips broken:** Channels with not dropped DiTis are marked as broken and are not used further in the pipetting sequence.
- **Abort Run:** The whole script is aborted.

14.1.3.2 FCA error: DiTi has been lost

Error received if one or more DiTis were lost during the execution of a script.

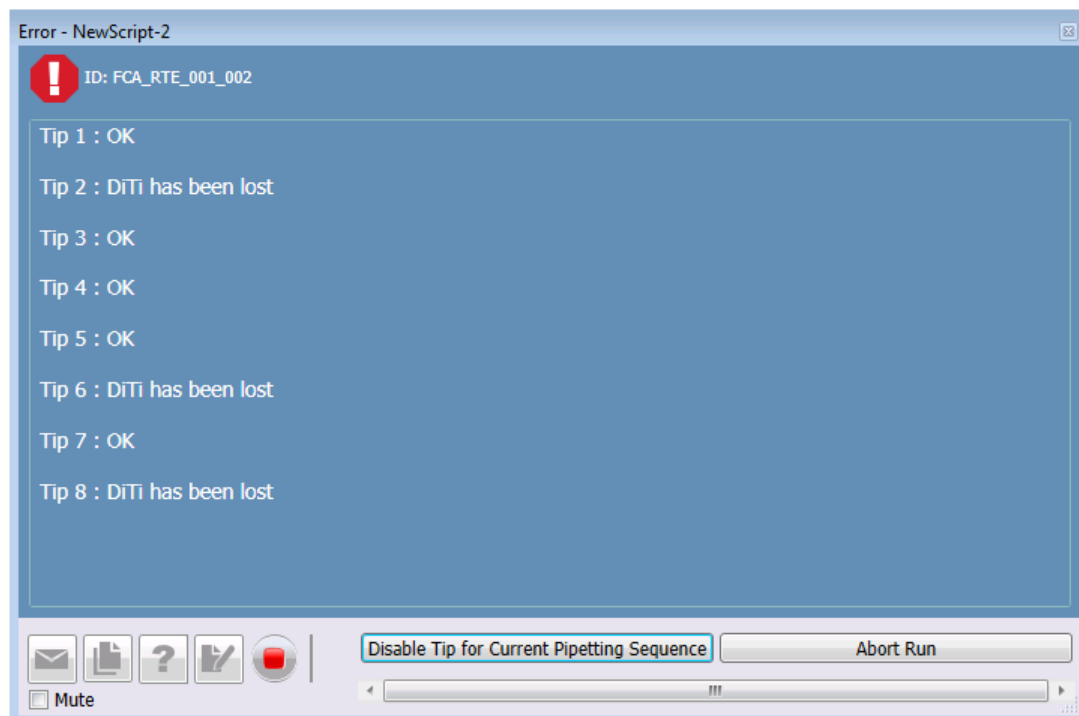


Fig. 168: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Disable Tip for Current Pipetting Sequence:** Channel(s) with lost DiTis are disabled for the current pipetting sequence and the execution of the script continues.
- **Abort Run:** The whole script is aborted.

14.1.3.3 FCA error: DiTi(s) not fetched

Error received if one or more DiTis were not properly fetched. The worktable is accessible for the user to fill the DiTi rack with new DiTis or to replace the whole DiTi rack.



When replacing the whole DiTi rack, put the new DiTi rack on exactly the same worktable position.

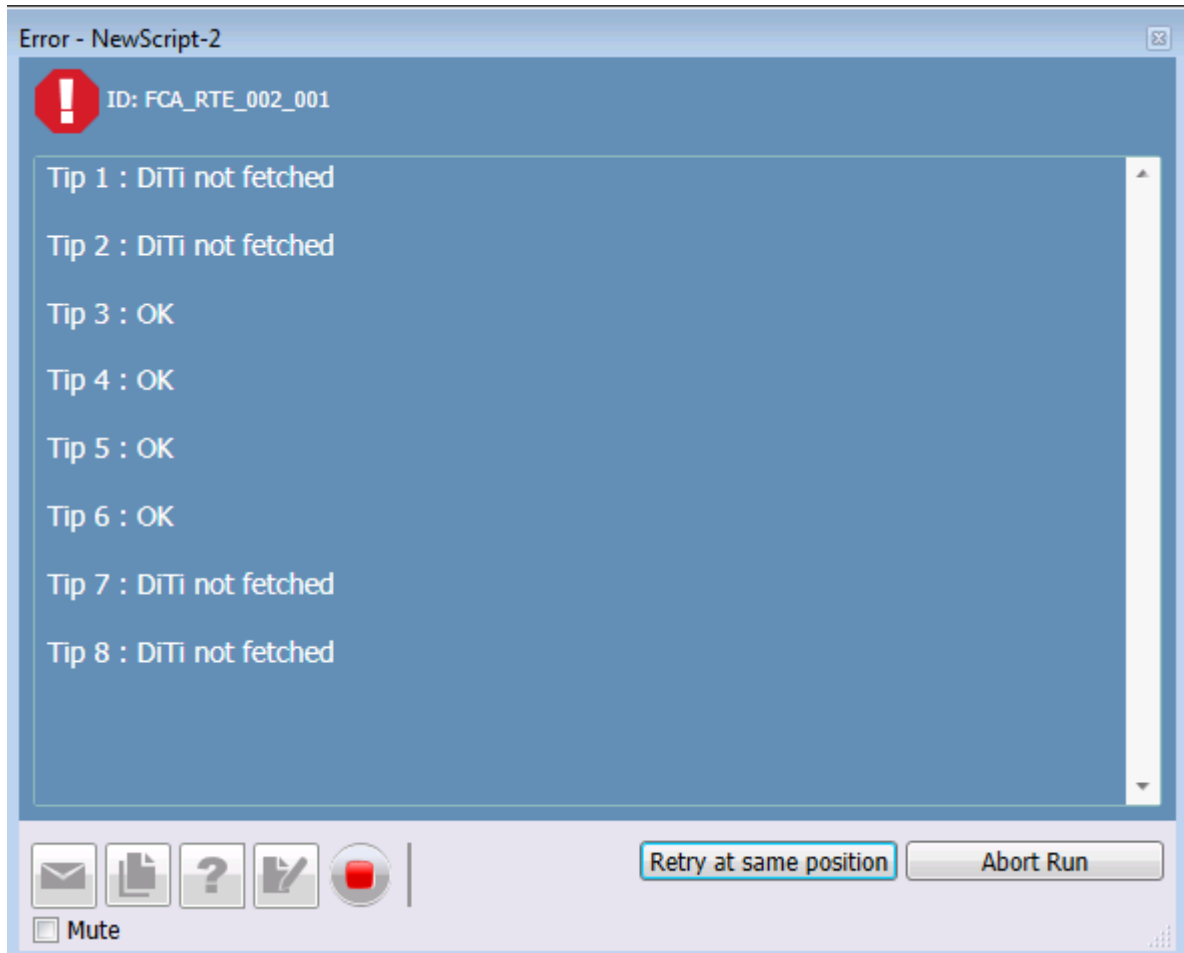


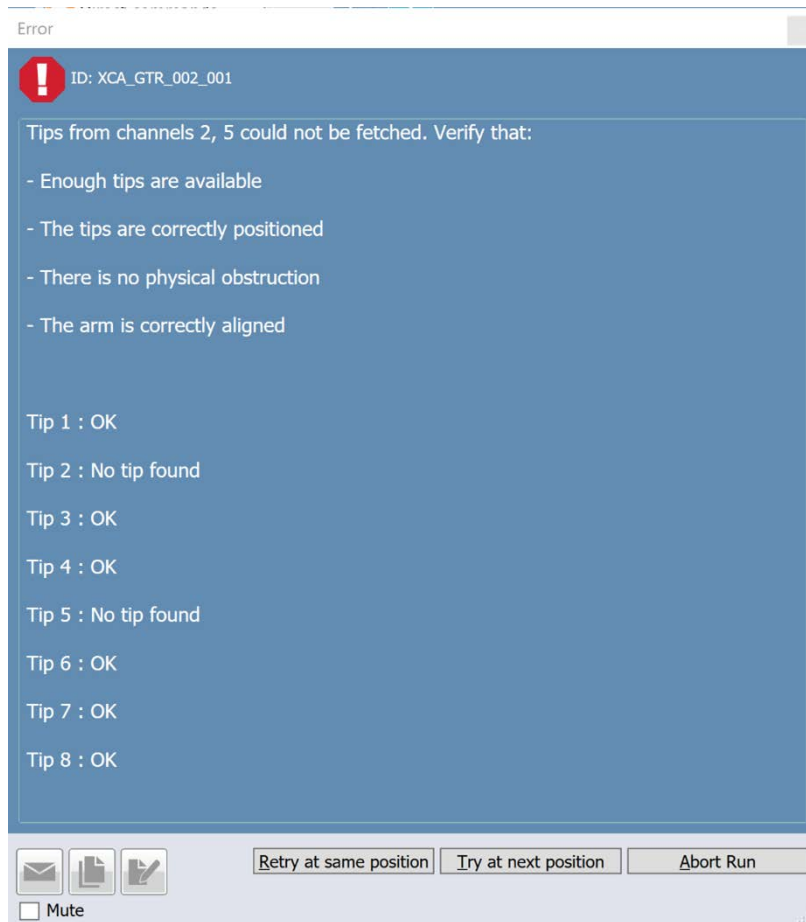
Fig. 169: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry at same position:** The FCA tries to get DiTis again from the same position as before.
- **Abort Run:** The whole script is aborted.

14.1.3.4 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

Abort Run: The entire run is aborted.

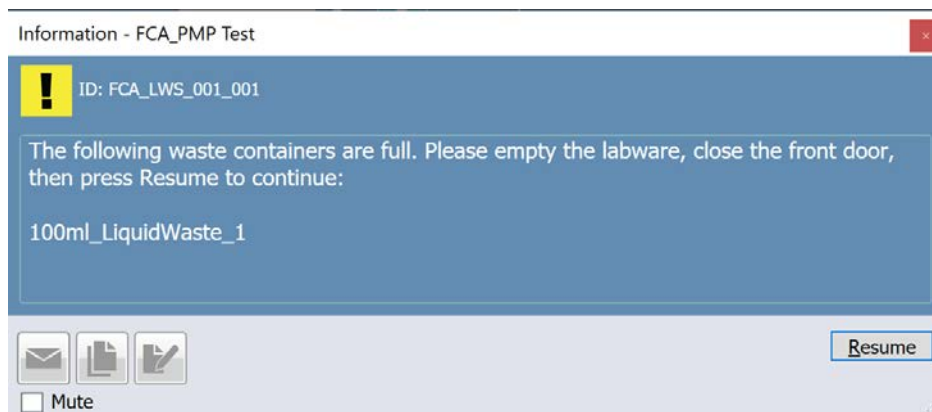
14.1.4 FCA Error: Liquid Waste Container Full

The software allows tracking the remaining capacity of a liquid waste, so as to prevent a possible overflow when dispensing liquid to the waste.

To activate this feature, the following steps are required:

- Define the usable capacity of the waste in microliters by setting the custom attribute LiquidWasteCapacity.
- Select the option "Use next when full" in the relevant Dispense and Empty Tips commands.

The software will then automatically switch to the next labware in the group until all labware items in the group are full. In that situation, the following prompt will appear:



Upon pressing Resume, the software resets the current volume of the labware to 0.

14.1.4.1 FCA error: DiTi(s) not dropped

Error received if one or more DiTis were not properly ejected.

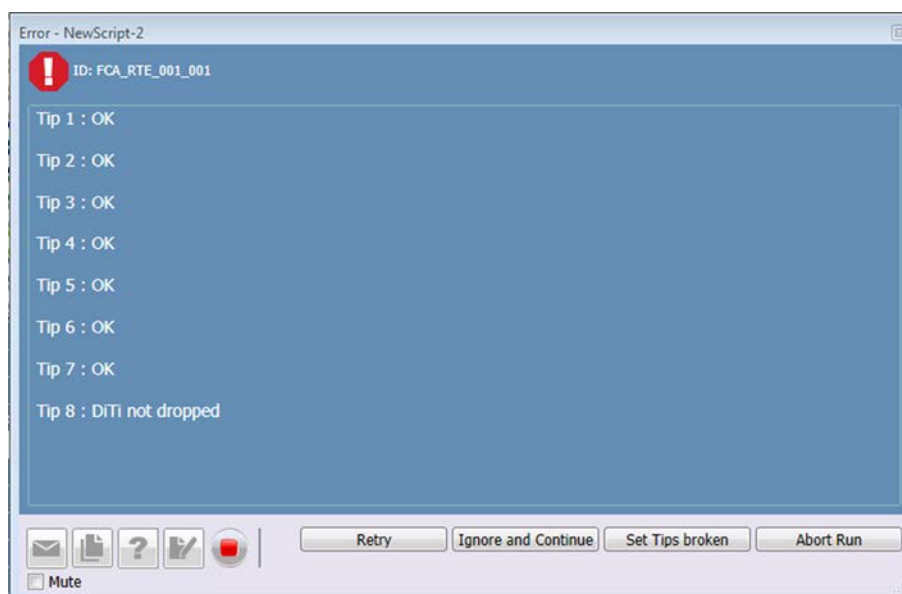


Fig. 170: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry:** The Drop Tips command executes again trying to eject the tips.
- **Ignore and Continue:** vControl continues the execution of the script regardless of whether the tips have been properly ejected.
- **Set Tips broken:** Channels with not dropped DiTis are marked as broken and are not used further in the pipetting sequence.
- **Abort Run:** The whole script is aborted.

14.1.4.2 FCA error: DiTi has been lost

Error received if one or more DiTis were lost during the execution of a script.

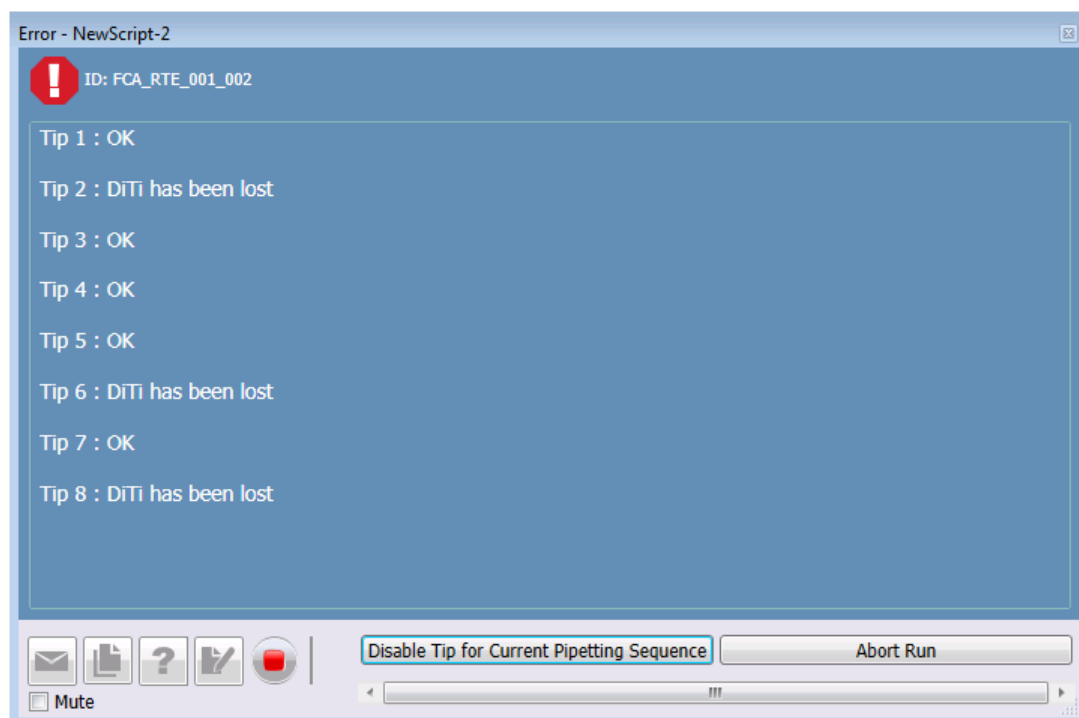


Fig. 171: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Disable Tip for Current Pipetting Sequence:** Channel(s) with lost DiTis are disabled for the current pipetting sequence and the execution of the script continues.
- **Abort Run:** The whole script is aborted.

14.1.4.3 FCA error: DiTi(s) not fetched

Error received if one or more DiTis were not properly fetched. The worktable is accessible for the user to fill the DiTi rack with new DiTis or to replace the whole DiTi rack.



When replacing the whole DiTi rack, put the new DiTi rack on exactly the same worktable position.

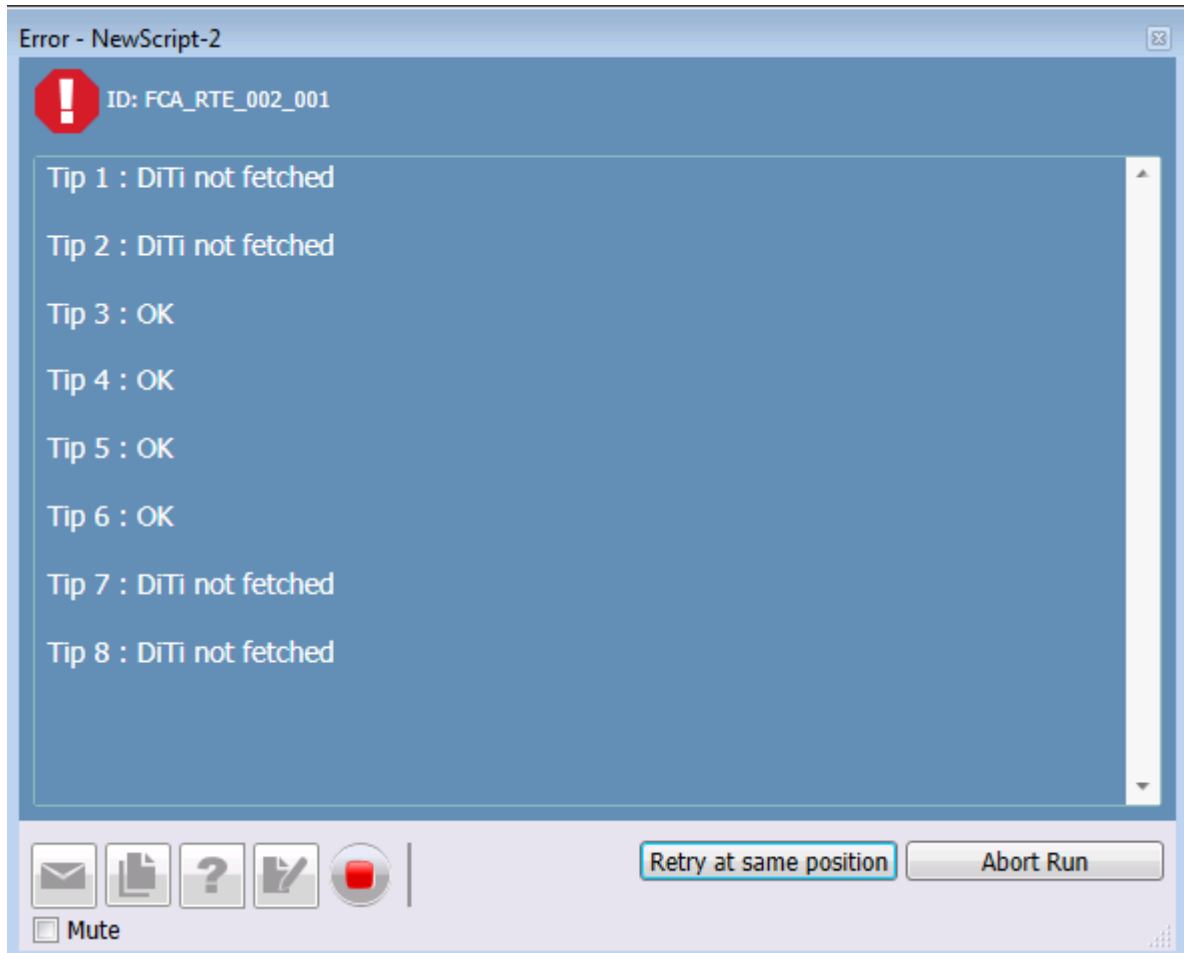


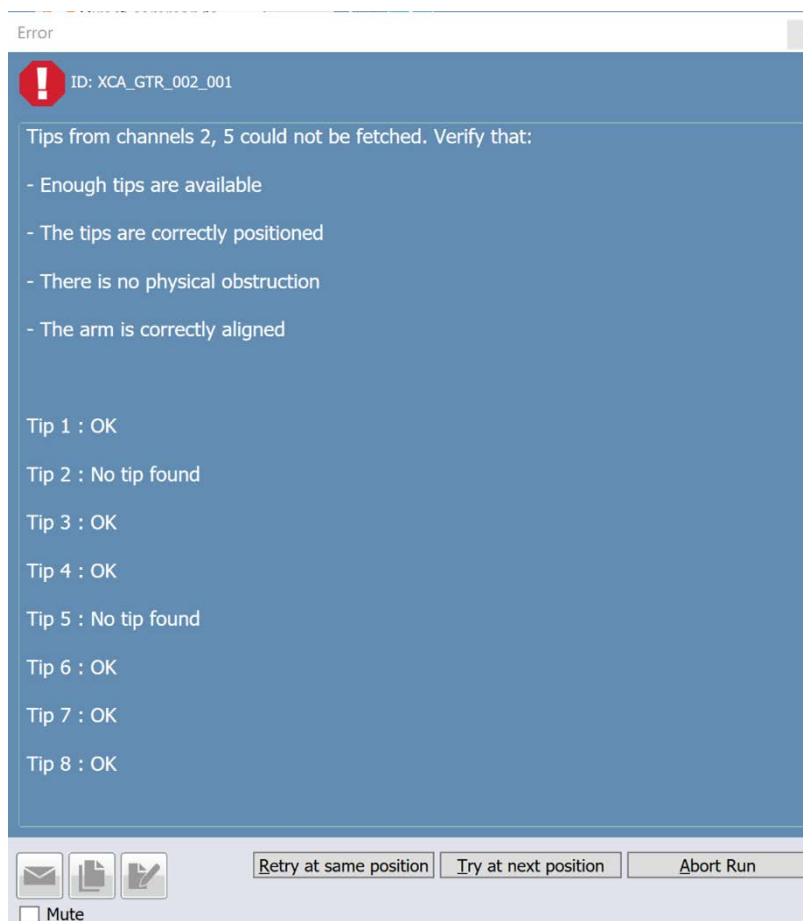
Fig. 172: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry at same position:** The FCA tries to get DiTis again from the same position as before.
- **Abort Run:** The whole script is aborted.

14.1.4.4 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

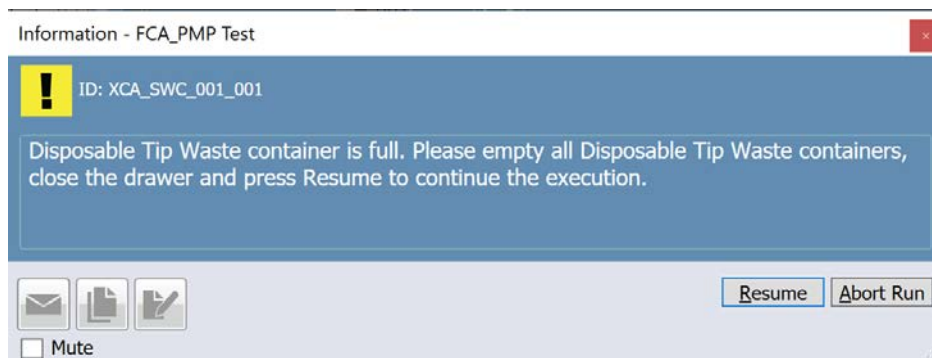
Abort Run: The entire run is aborted.

14.1.5 FCA Error: Disposable Tip Waste Full

The software tracks the remaining capacity of the disposable tip wastes, so as to notify the user on time to empty the waste containers and prevent the accumulating tips from obstructing further discards. This feature uses two custom attributes:

- **SolidWasteCapacity** set for the waste labware, which defines the capacity of the tip waste in “waste units”.
- **SolidWasteConsumption** set for each tip type, which defines the waste consumption of one tip in “waste units”.

If the waste capacity has been reached after a Drop Tips command, the software prompts the user to empty the waste container:



14.1.5.1 FCA error: DiTi(s) not dropped

Error received if one or more DiTis were not properly ejected.

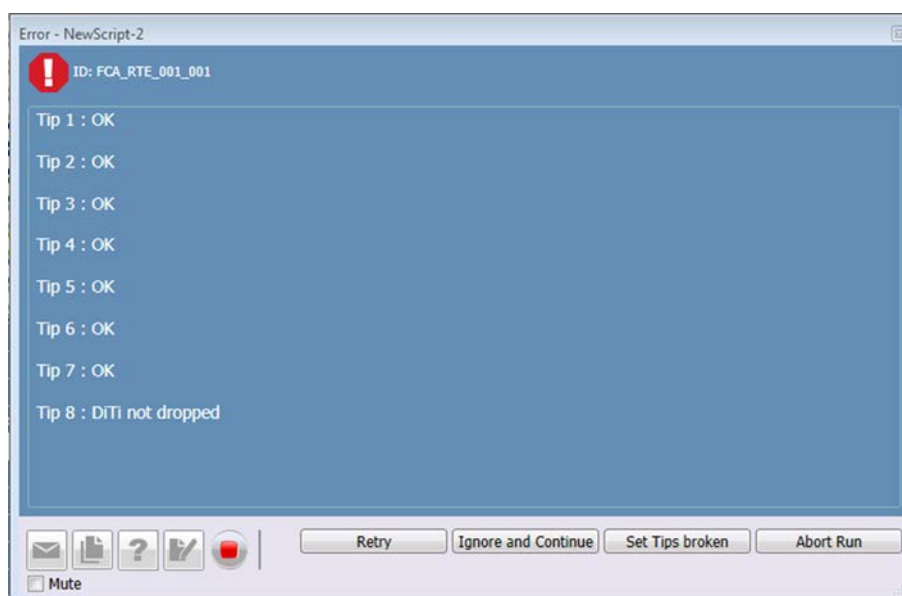


Fig. 173: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry:** The Drop Tips command executes again trying to eject the tips.
- **Ignore and Continue:** vControl continues the execution of the script regardless of whether the tips have been properly ejected.
- **Set Tips broken:** Channels with not dropped DiTis are marked as broken and are not used further in the pipetting sequence.
- **Abort Run:** The whole script is aborted.

14.1.5.2 FCA error: DiTi has been lost

Error received if one or more DiTis were lost during the execution of a script.

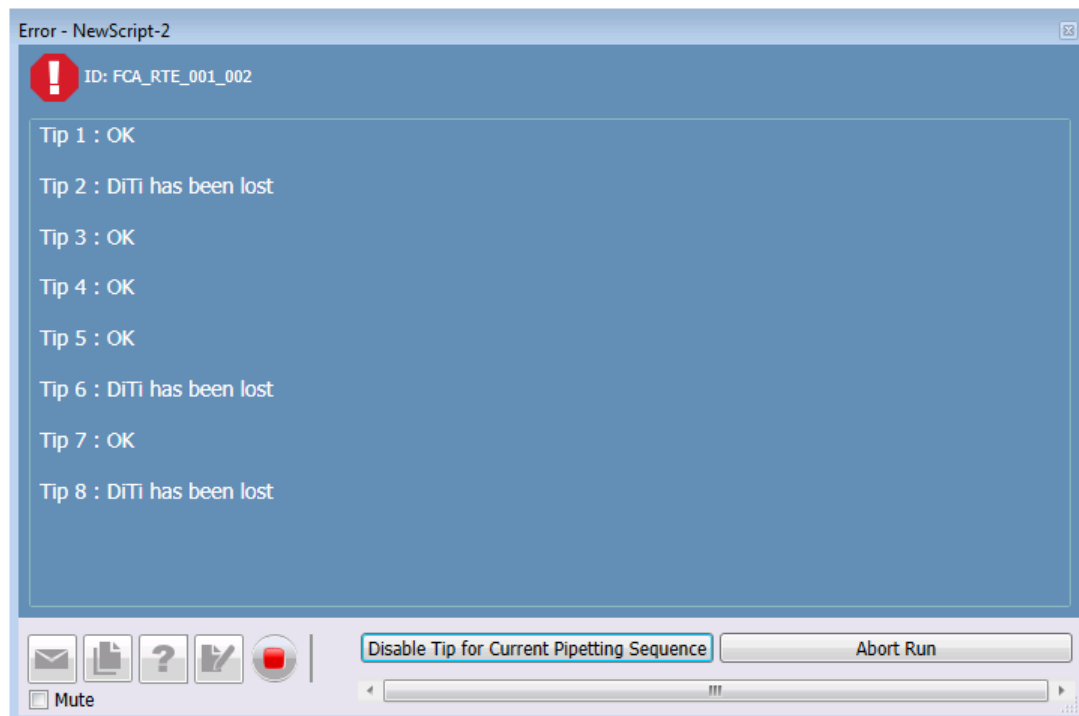


Fig. 174: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Disable Tip for Current Pipetting Sequence:** Channel(s) with lost DiTis are disabled for the current pipetting sequence and the execution of the script continues.
- **Abort Run:** The whole script is aborted.

14.1.5.3 FCA error: DiTi(s) not fetched

Error received if one or more DiTis were not properly fetched. The worktable is accessible for the user to fill the DiTi rack with new DiTis or to replace the whole DiTi rack.



When replacing the whole DiTi rack, put the new DiTi rack on exactly the same worktable position.

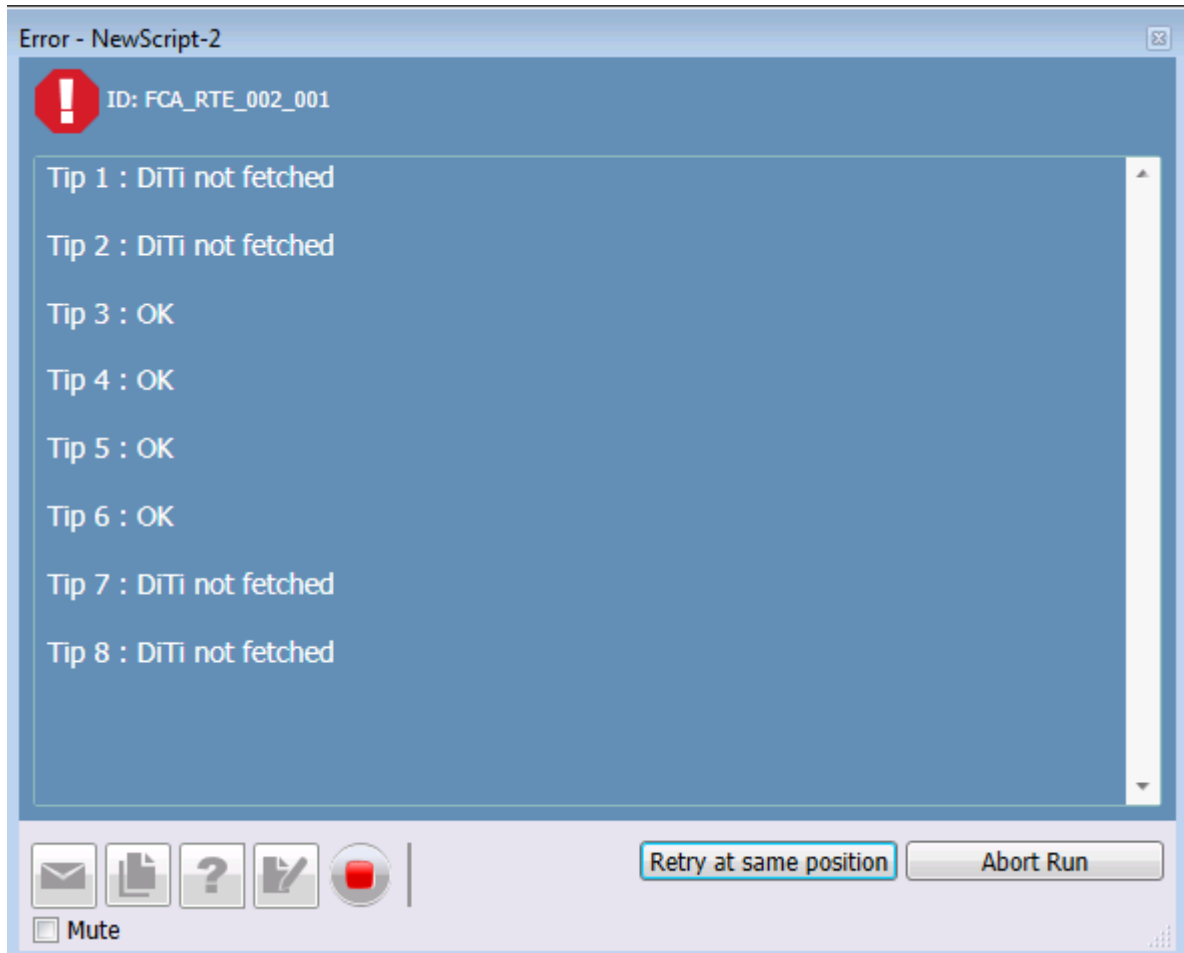


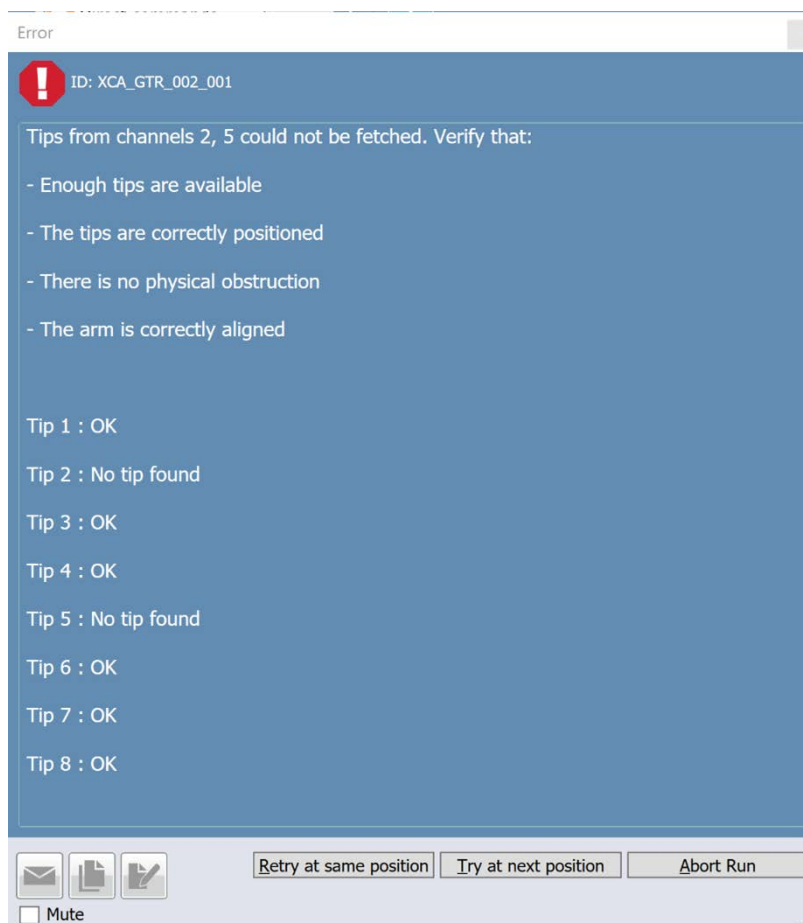
Fig. 175: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Retry at same position:** The FCA tries to get DiTis again from the same position as before.
- **Abort Run:** The whole script is aborted.

14.1.5.4 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

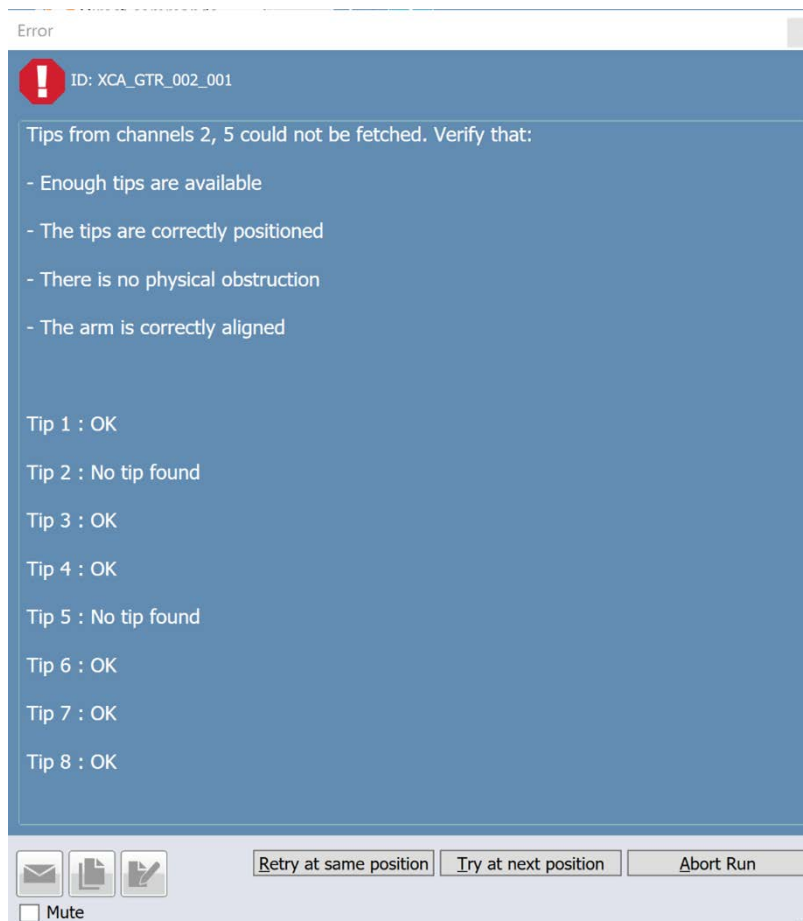
Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

Abort Run: The entire run is aborted.

14.1.6 FCA error: DiTi(s) not fetched

This error message is shown if one or more disposable tips were not properly fetched. The worktable is accessible for the user to fill the tip box with new DiTis or to replace the whole box.



The user has the following options to handle the error:

Retry at same position: The FCA tries to get DiTis again from the same position as before.

Try at next position: The FCA moves to the next available position and tries to fetch tips.

Abort Run: The entire run is aborted.

14.2 Error without User Interaction (FCA)

This chapter describes runtime error messages and the associated dialog boxes that can be issued by the software. It gives information on the cause of the error and instructs how to correct the error, where applicable.

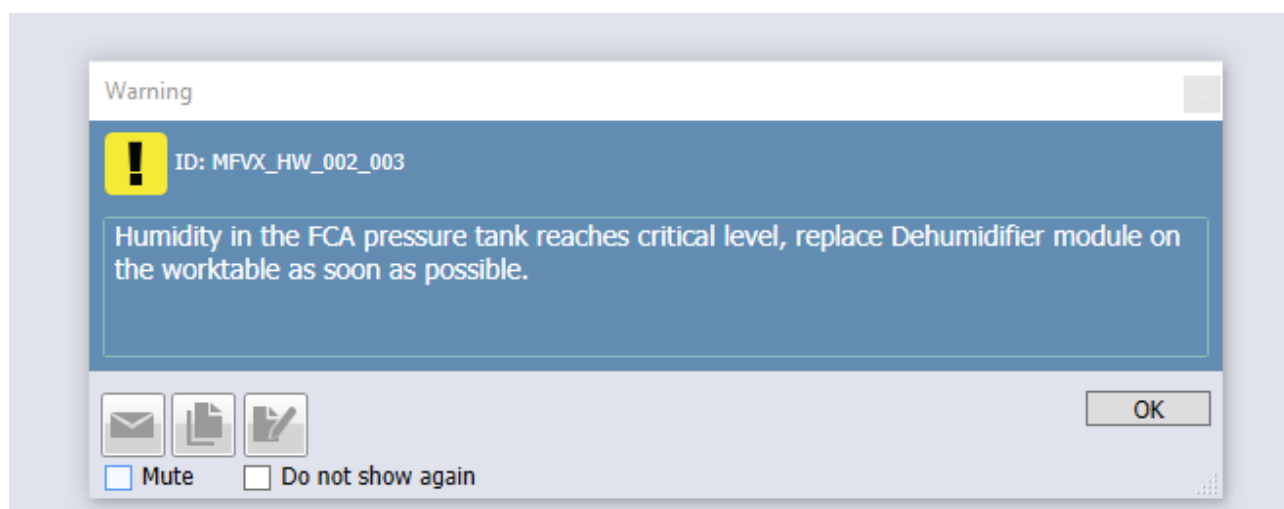
14.2.1 Dehumidifier Notification

Warning

The ARP humidity controlling system sends a warning message in the System Trace View section in case the humidity in the tank reaches a critical level. The warning message appears once during instrument initialization or at the beginning of the script execution.

System Trace View						
Status	Start Time	End Time	Module	Macro	Object	Message
✓	08:55:52	08:55:52	SystemSW-3600I	Initialization		Instrument initialization finished.
⚠	08:55:57	08:55:57	SystemSW-3600I			Humidity in the FCA pressure tank reaches critical level, replace Dehumidifier module on the worktable as soon as possible.
✓	08:55:52	08:55:52	SystemSW-3600I	Initialization		Driver initialization finished.

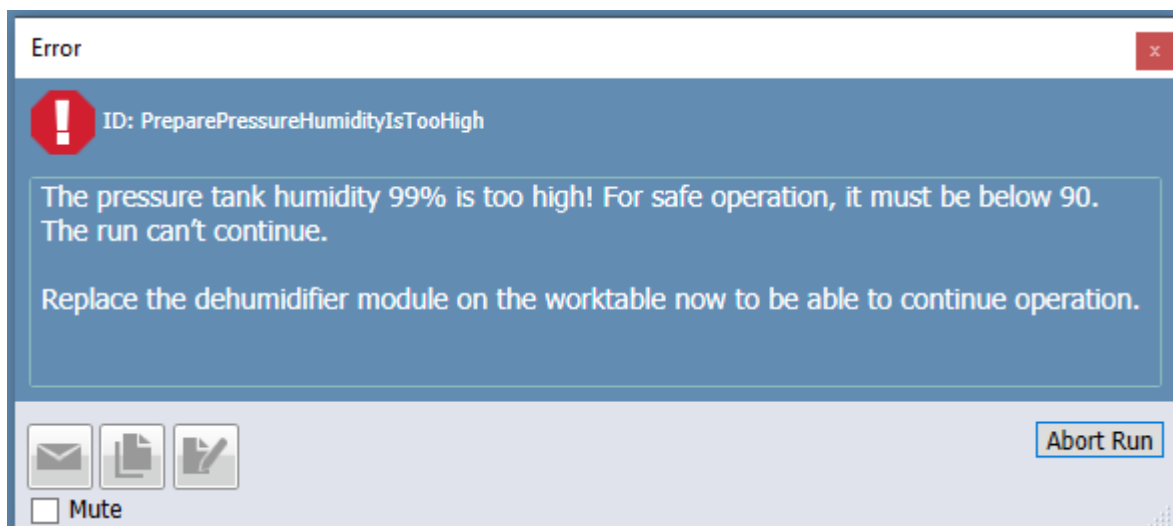
If the warning message was ignored and no actions to replace the dehumidifier system were taken, a warning message will again appear at the end of the script execution, which will also be sent to Introspect.



Error

Error If the ARP system reports an unacceptable humidity level (here 99%), which will damage hardware eventually, an error is reported and operation can't be continued. In this case immediate replacement of the dehumidifier module is required. Both a system trace entry and a dialog appears and a run has to be aborted.

System Trace View						
Status	Start Time	End Time	Module	Macro	Object	Message
✗	09:07:53	09:07:53	SystemSW-3600I			The pressure tank humidity 99% is too high! For safe operation, it must be below 90. The run can't continue. Replace the dehumidifier module on the worktable now to be able to continue operation.



14.3 Error Requiring User Interaction (RGA)

This section describes pipetting and hardware device error messages which require user interaction. Please see the description of the output file format from the Export Data command for information on the pipetting error codes which are reported. Errors are also reported in the vControl log files.

- RGA Error: Labware lost
- RGA Error: Labware not gripped on source location
- RGA Error: Object not gripped on source location

14.3.1 RGA error: Labware lost

Error received if the labware being transferred has been lost.

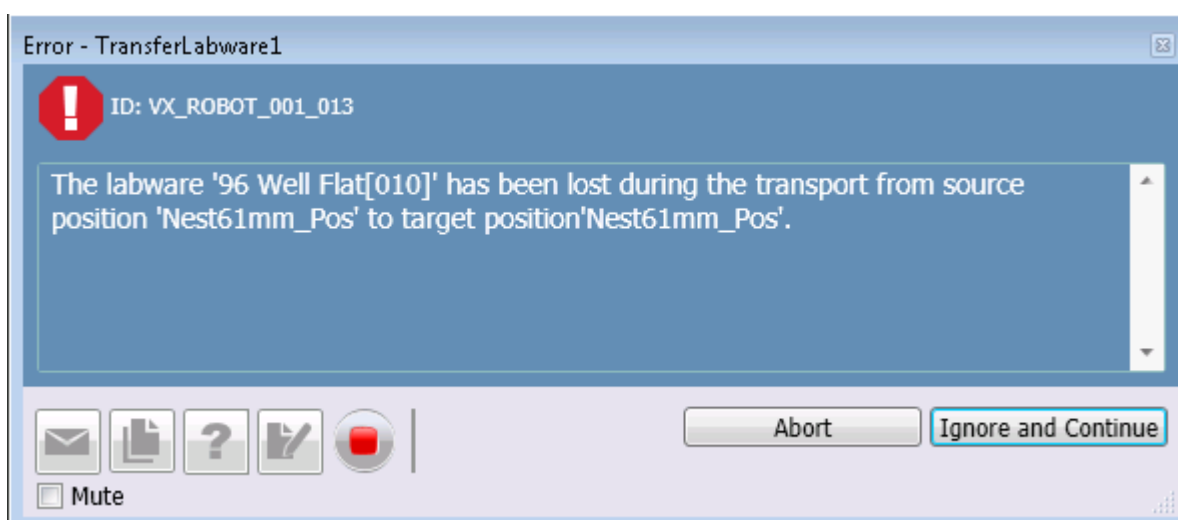


Fig. 176: Error message

When receiving this error message, the user has the following options to handle the error properly:

- **Abort Run:** The whole script is aborted.
- **Ignore and Continue:** vControl continues the execution of the script ignoring that the labware was lost.

14.3.2 RGA error: Labware not gripped on source location

Error received if the labware was not successfully gripped at the source location.

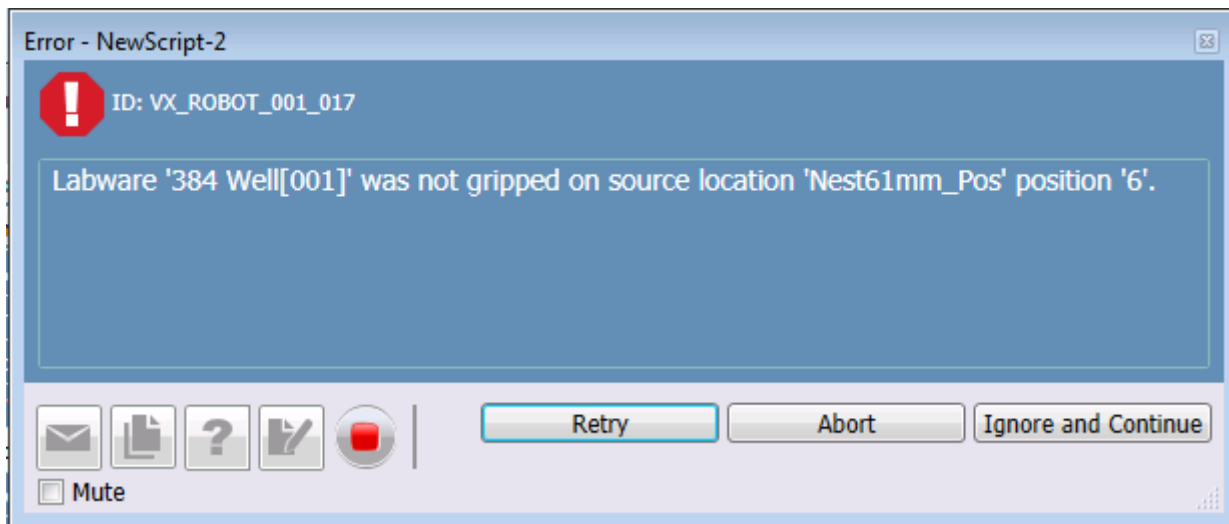


Fig. 177: Error message

At this point, the worktable is accessible to the user. When receiving this error message, the user has the following options to handle the error properly:



*Before clicking **Retry**, the user can access the worktable and place the object on the source position.*

- **Retry:** The RGA tries to grip the labware from the same position as before.
- **Abort:** The whole script is aborted.
- **Ignore and Continue:** vControl continues the execution of the script ignoring that the labware was not fetched.

14.3.3 RGA error: Object not gripped on source location

Error received if the object was not successfully gripped at the source location.



Fig. 178: At this point, the worktable is accessible to the user

When receiving this error message, the user has the following options to handle the error properly:



*Before clicking **Retry**, the user can access the worktable and place the object on the source position.*

- **Retry:** The RGA tries to grip the object from the same position as before.
- **Abort:** The whole script is aborted.
- **Ignore and Continue:** vControl continues the execution of the script ignoring that the labware not fetched.

14.4 Error Requiring User Interaction (Loading ID)

This section describes errors requiring user interaction using the LoadingID. Each sub-chapter explains an error type and ways how to recover from an error.



If the detection sequence is wrong, the system does not accept the tube runner and informs the user.

The software only accepts barcodes from the firmware if:

- Barcode could be read
- Checksum is correct
- Barcode type is allowed
- Detection sequence is correct (checked by firmware, no software needed):
 - Runner Start barcode read
 - Tube Barcodes read
 - Runner Endbarcode read
 - Hall Sensor detects contact

14.4.1 Unreadable Tube Barcode

When receiving this error message, the user has the following options to handle the error properly:

- Check barcode label and position on tube.

- Reload with slower move.
- Check allowed barcode configuration in settings.
- Clean window of the scanner.

14.4.2 Unreadable Runner Barcode

When receiving this error message, the user has the following options to handle the error properly:

- Check runner barcode.
- Check runner type.



Fig. 179: No barcode found error

14.4.3 Runner Not Detected (Hall Sensor)

When receiving this error message, the user has the following options to handle the error properly:

- Check runner type (Loading ID runner).

14.4.4 Runner Not Defined in Worktable

When receiving this error message, the user has the following options to handle the error properly:

- Loaded wrong runner type according to worktable, not defined runner, on wrong position.

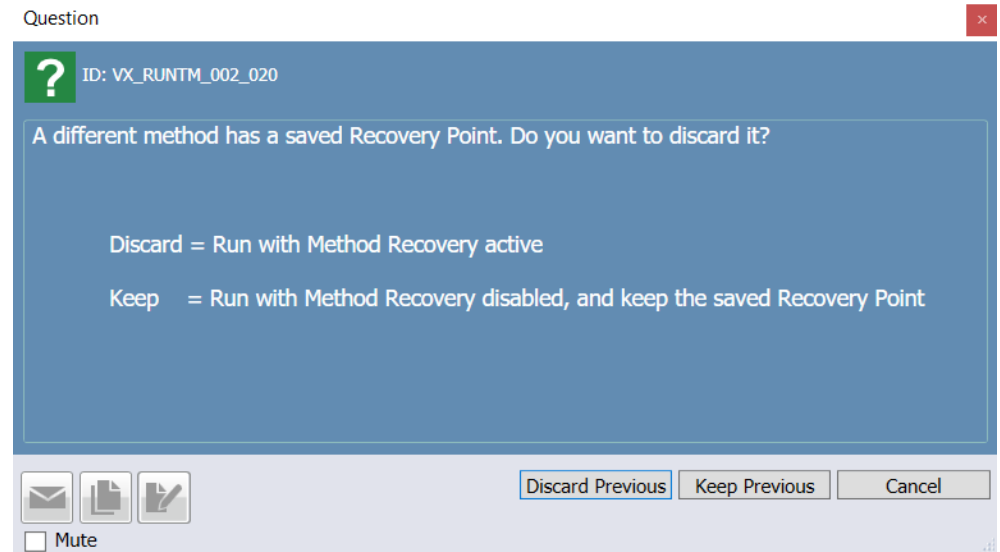
14.4.5 Continue Not Allowed after Loading

When receiving this error message, the user has the following options to handle the error properly:

- Check definition in the **Start Loading** statement.

15 Method Recovery

Method recovery is a function that (if configured) enables the recovery of individual script commands if a method has been canceled either manually or due to an error or power failure, etc.



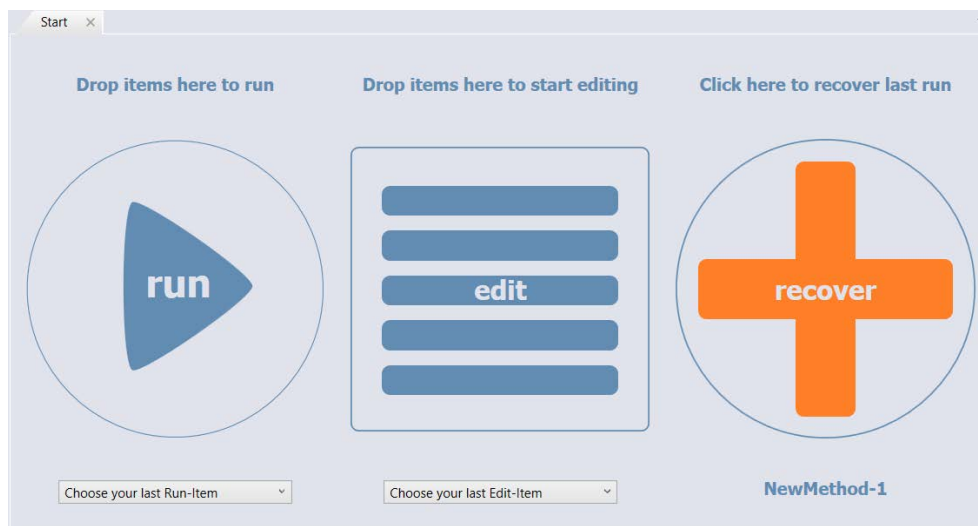
The vControl recovery repeats or skips the command that was being executed when the method was canceled. Depending on the command, there are several scenarios where the software will either perform an automatic retry when the restore is started or the user will be asked what to do at the beginning of the restore execution.

15.1 How to Setup and Run Method Recovery

Method recovery is enabled by default, but can be disabled under **Settings -> Configure system -> Application settings -> Disable Method Recovery**. If this is configured, the software creates recovery points for certain actions, e.g. when Stop or Abort is pressed or if the software is shut down unexpectedly. The recovery point is then displayed on the vControl start page. Clicking on recover loads the **Recovery Editor**.

15 - Method Recovery

How to Setup and Run Method Recovery

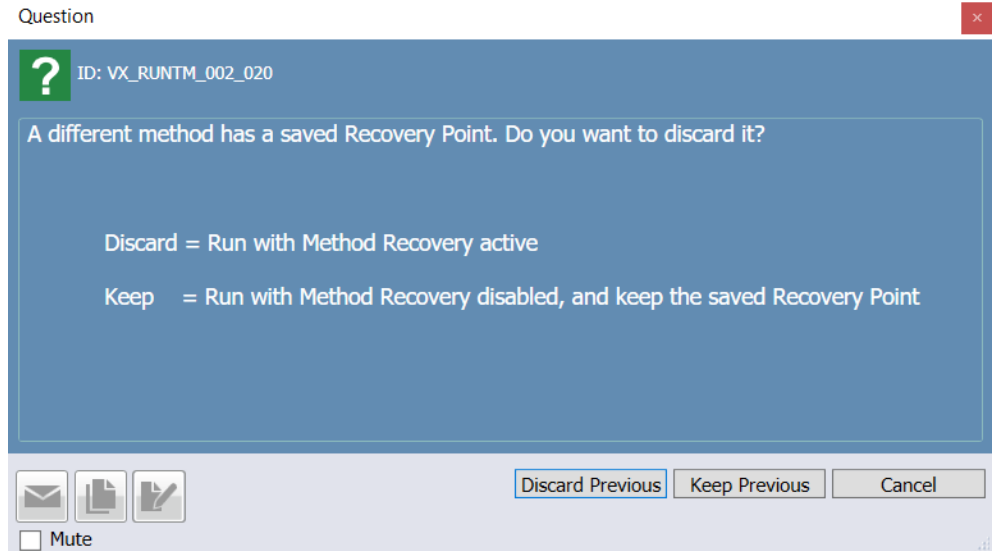


The recovery editor allows the user to recover the method from the point at which it failed. Commands shown in green have been executed successfully, commands shown in blue have not yet been executed. The orange command is the point at which the execution of the method failed. Pressing Play recovers the command to the point at which it failed.



15.1.1 Method Recovery

Method recovery is a function that (if configured) enables the recovery of individual script commands if a method has been canceled either manually or due to an error or power failure, etc.



The vControl recovery repeats or skips the command that was being executed when the method was canceled. Depending on the command, there are several scenarios where the software will either perform an automatic retry when the restore is started or the user will be asked what to do at the beginning of the restore execution.

15.1.2 Bringing the Instrument to a defined State

It is possible that after a method error the device is not in the required state to recover immediately and continue the run. An example of this is a power failure during an aspiration step. After a power failure, the device must be initialized, and this can only be done after the tips have been manually removed from the channels. If the device is still initialized, the tips can be removed with direct commands before the method is restored. The user is responsible for bringing the instrument to a state that corresponds to the restore point, e.g. by placing tips using the **get tips** direct commands.

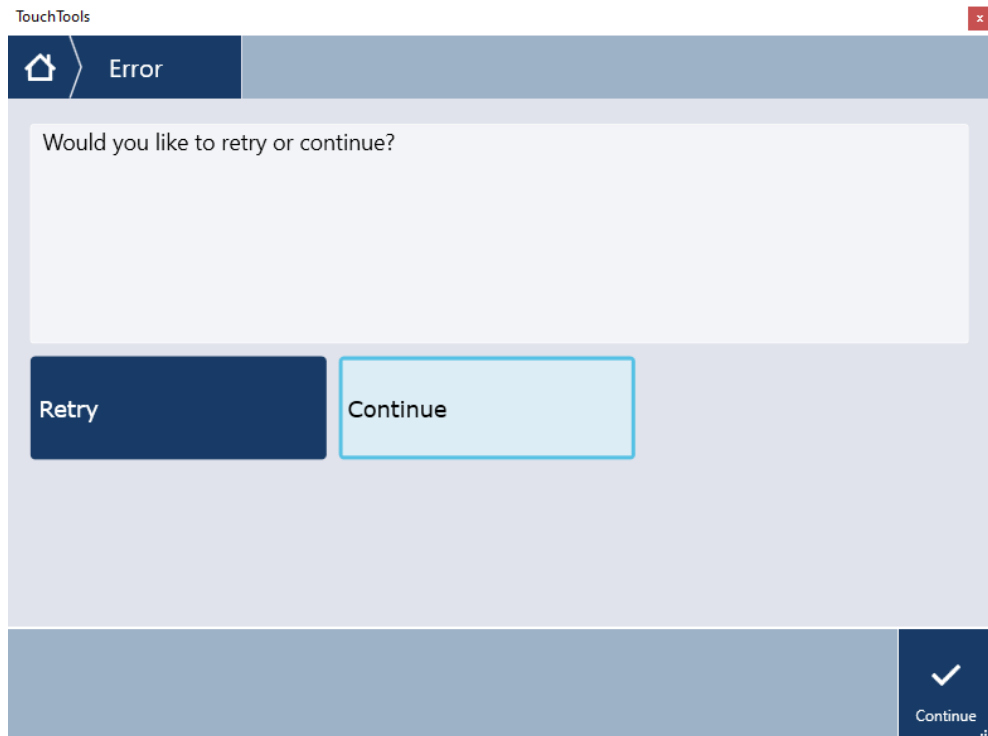
15.2 Script Commands

The recovery of a script command is handled within the script command itself. In the recovery phase, the user can choose how the command should be recovered. A generic restore is offered for commands that are either to be retried or continued. A defined, specific recovery is offered for commands that require further processing to reach a specific state.

15.2.1 FCA Commands – Generic Recovery

Commands that do not require any special handling to return to a defined state after the recovery offer a **Retry** / **Continue** option.

When a command is recovered, the following user interface is displayed. If you click on **Retry**, the entire command is executed again. With **Continue**, the command is considered completed and continues to the next one. The text displayed is customized and is described below for the individual commands.



15.2.1.1 Drop Tips

Text: Drop Tips has been recovered. Please retry dropping all tips or continue to the next command.

Options:

- **Retry:** Re-executes Drop tips with all channels.
- **Continue:** Assumes that all tips have been dropped and moves on to the next command.

15.2.1.2 Get Fingers

Text: Get Fingers has been recovered. Please retry picking up the channel grippers or continue to the next command. When retrying, please unmount the channel grippers.

Options:

- **Retry:** Re-executes get fingers.
- **Continue:** Assumes that the channel grippers are mounted correctly and moves on to the next command.

Note: If the command is tried again

15.2.1.3 Drop Fingers

Text: Drop Fingers has been recovered. Please try dropping the channel gripper again or continue with the next command.

Options:

- **Retry:** Re-executes get fingers.
- **Continue:** Assumes that the fingers have been dropped and continues with the next command..

15.2.1.4 Transfer Labware

Text: The transfer labware has been restored. Please try again to pick up the labware at the source location or manually place the labware at the destination location and continue with the next command.

Options:

- **Retry:** Executes the entire Transfer Labware again.
- **Continue:** Assumes that the labware has reached the target and moves on to the next command.

Clicking on either option will display a prompt asking the user to return the labware to either the source location or the destination location. The prompt contains a **Next** button that must be pressed once the action is complete.

15.2.1.5 Move Arm

Text: None

Try again immediately, as it makes no sense to skip the Move Arm. If the execution has failed, it must be repeated. As soon as the movement is completed, the next command is executed immediately.

15.2.1.6 Empty Tips

Text: Empty tips has been recovered. Please retry emptying the tips or continue to the next command.

Options:

- **Retry:** Re-Executes the entire empty tips.
- **Continue:** Assumes that the tips have been emptied and will move to the next command.

15.2.1.7 Channel Self Test

Text: None.

Try again immediately, as it makes no sense to skip the channel self-test. If it fails during execution, it must be repeated. As soon as the test is completed, the system immediately moves on to the next command.

15.2.1.8 Set Pipetting Error Handling

Text: None.

Try again immediately, as it makes no sense to skip the pipetting error handling. If the error handling setting has failed, it must be carried out again. As soon as everything has been set, it immediately continues to the next command.

15.2.1.9 Start Prepare Pressure

Text: None.

Try again immediately, as it makes no sense to skip the preparation. If the command failed during preparation, it must be executed again. As soon as everything is completed, the next command is executed immediately.

15.2.1.10 Switch Off PMP

Text: None.

Repeat immediately, as it makes no sense to skip switching off. If the command failed when switching off, it must be executed again. If it was successful, the next command is executed immediately.

15.2.1.11 Switch On PMP

Text: None.

Repeat immediately, as it makes no sense to skip switching on. If the command failed when switching on, it must be executed again. If it was successful, the next command is executed immediately.

15.2.2 FCA Commands – Detailed Recovery

To achieve a defined state, some FCA commands require several recovery steps. These are mainly the pipetting commands

15.2.2.1 Get Tips

Text: Get Tips has been recovered, please try to pick up the tips again or continue with the next command. The currently mounted tips are:

Channel 1: FCA 200ul

Channel 2: FCA 200ul

Channel 3: None

Channel 4: None

Options:

- **Drop Tips and Retry all Channels:** Executes a Drop Tips before re-executing Get Tips with all channels.

- **Retry failed Channels:** Re-executes Get Tips with the channels which don't have tips mounted (according to the MAP driver state).

- **Continue:** Assumes that all tips have been mounted and moves on to the next command.

Get Tips ADH recovery:

- The recovery will recover the command that was executed in the ADH Subroutine during an abort.

15.2.2.2 Detect Liquid

Text: Detect Liquid has been recovered. Please retry the liquid level detection or continue to the next command.

- **Retry with Current Tips:** Re-executes the entire command, without dropping the tips.

- **Retry with New Tips:** Executes a drop tips command and mounts new tips of the same type. After this, the command is re-executed.

- **Continue:** Assumes the liquid level detection was successful and moves on to the next command.

15.2.2.3 Aspirate

An Aspirate Script Command displays two user prompts. The first one defines what shall be done with the tips. The second user prompt defines what shall be done with the liquid which might still be inside the tip.

User Prompt 1:

Text: Aspirate has been recovered. Please retry the Aspiration or continue to the next command.

Options:

- **Retry with Current Tips:** Re-executes the entire command.

- **Retry with New Tips:** Executes a drop tips command and mounts new tips of the same type. After this, the command is re-executed.

- **Continue:** Assumes the aspiration was successful and moves on to the next command.

User Prompt 2: This user prompt is only shown when a retry option from the first user prompt is selected.

Text: Press 'Drop Tips to Waste' to discard the tips directly to waste, or press 'Back to Source' if you want to dispense the tip content to the source position before discarding the tips.

Options:

- **Back to Source:** An empty tips command will be executed into the Source from where the liquid was aspirated.

- **To Waste:** An empty tips command will be executed into the liquid waste if available, if the liquid waste is not available:

If the option **Retry with current Tips** is selected, **Text: There is no available Liquid Waste labware on the worktable. Please dispense the entire liquid volume back to the source location or abort the recovery.** A recovery options will be **Back to Source** and **Abort Recovery**.

If the option **Retry with new Tips** is selected, the tips will be dropped into the tip waste without dispensing the liquid.

- **Abort:** This aborts the recovery.

15.2.2.4 Dispense

Text: Dispense has been recovered. Please retry to dispense the full volume or continue to the next command.

Options:

- **Retry:** The entire volume will be re-dispensed into the destination well.

- **Continue:** Assumes the dispense was successful and moves on to the next command.

Note: The sequence of **Move to Z-Dispense**, **Dispense**, **Move to Z-Travel** are considered as one step and will finish executing when pressing abort. A state in between these steps can only be achieved through a loss of power, a collision, or a software crash.

A multi dispense is split up into multiple dispense script commands within the script. If a dispense fails, the individual dispense step is recoverable. After this has been recovered, the remaining steps will be executed.

15.2.2.5 Mix

Text: Mix has been recovered. Please retry to Mix the entire liquid or continue to the next command.

Options:

- **Retry:** The entire volume is mixed again, all cycled are repeated.

- **Continue:** Assumes the mix was successful and moves on to the next command.

Note: The sequence of **Move to Z-Start**, **Mix (Aspirate, Dispense x cycles)**, **Move to Z-Travel** are considered as one step and will finish executing when pressing abort. A state in between these steps can only be achieved through a loss of power, a collision, or a software crash.

15.2.3 FCA Commands – Smart Commands

A Smart Command generates a subscript that is executed when the step of the smart command in the script is reached. The subscript consists of script commands, which are described above. There is no special recovery function for Smart Commands.

15.2.4 RGA Commands

15.2.4.1 Transfer Labware

User Prompt 1

Text: The system has detected a gripped labware. The labware must be removed from the grippers. Move the RGA to a position where it can release the gripped labware and then click **Open Grippers** when you are ready to release the labware. The grippers will open 10 seconds after the button is pressed. The grippers will open 10 seconds after pressing the button.

Options:

- **Open Grippers:** The grippers will open after 10s

User Prompt 2:

Text: Labware will be released in 10 seconds or by clicking the OK button, the grippers will immediately open.

Options: - **OK:** The grippers open.

User Prompt 3

Text: Move the labware '{microplateName}' in Nest7mm_Pos to position 2, close the front door and confirm the action to execute the command.

Options:

- **Confirm:** The run continues

15.2.4.2 Move Arm & Execute Single Vector

Text: None.

Retry immediately, as it does not make sense to skip the command. If it failed during the execution, it must be repeated. Once the move is complete it moves immediately to the next command.

15.2.5 Inheco Commands – MTC and InhecoIncubator

15.2.5.1 Enable Temperature Control

Text: None.

Try again immediately, as it makes no sense to skip the command. If it has failed during execution, it must be repeated. As soon as the movement is completed, the next command is executed immediately.

15.2.5.2 Wait for Temperature

Text: None.

Try again immediately, as it makes no sense to skip the command. If it has failed during execution, it must be repeated. As soon as the movement is completed, the next command is executed immediately.

15.2.5.3 Disable Temperature Control

Text: None.

Try again immediately, as it makes no sense to skip the command. If it has failed during execution, it must be repeated. As soon as the movement is completed, the next command is executed immediately.

15.2.5.4 Start Shaking

Text: Start shaking has been restored. Please try to start the shaking process again or continue with the next command.

Options:

- **Retry:** If the device is not shaking, a start shaking command is executed. The entire shaking procedure will be executed again. This is checked through the status of the MAP driver.

- **Continue:** Assumes that shaking is complete (or started) and will continue to the next command.

Note: If the device is shaking, then vControl automatically moves on to the next command.

15.2.5.5 Stop Shaking

Text: Stop Shaking has been recovered. Please retry stop shaking or continue to the next command.

Options:

- **Retry:** The command will attempt to stop the shaking again.

- **Continue:** Assumes that shaking has been stopped and continues to the next command.

Note: If the device is not shaking, then vControl automatically moves on to the next command.

15.2.6 Inheco Commands – ODTc

15.2.6.1 Close ODTc Door

Text: None

Retry immediately

15.2.6.2 Open ODTc Door

Text: None

Retry immediately

15.2.6.3 Upload ODTc Method Set

Text: None

Retry immediately. If the file already exists, the software will overwrite the file.

15.2.6.4 Start ODTc Method

Text: Start ODTc Method has been recovered. Please retry starting the ODTc method or continue to the next command.

Options:

- **Retry:** Will retry to start the ODTc Method.

- **Continue:** Assumes the method has been started and continues to the next command.

15.2.6.5 Wait for ODC Methods

Text: Wait for ODC Methods has been recovered. Please retry the wait, or continue to the next command.

Options:

- **Retry:** Will continue to wait for the ODC Methods.
- **Continue:** Assumes the method has been completed and continues to the next command.

16 API

The vControl API allows external applications and scripts to control vControl. It allows to run methods, read/write variables, stop/pause/resume a run, trigger execution of script commands and much more.

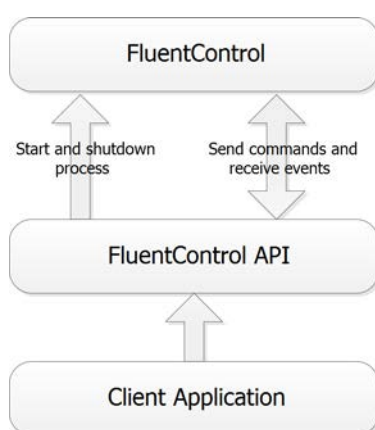


Fig. 180: Overview

Why to use the API?

vControl's main feature is to automate assay execution. However, to start and configure an assay run the user still needs to interact with the application:

1. Start a method.
2. Enter parameters (e.g., sample count, barcodes, etc.).
3. Monitor progress and react on errors.



The API allows for automating these steps with external applications and scripts.

Example

- Connect to a database or other external resources
An external application can read data from a database or from a web service and dynamically start a vControl method and parametrize the run.

16.1 Execution Channels

Execution Channels allow for triggering script command executions in vControl from an API client.

In vControl a **script with worktable** and a **Create API Channel** script command is required for creating an Execution Channel.

When **Create API Channel** is executed, a new subroutine is started. The event **ChannelOpens** assigns an Execution Channel instance that can now receive script commands via API.

The API client can now use the Execution Channel to send commands and to trigger **Abort** or **Finish** of the Execution Channel.

- **Abort:** The complete run will abort.
- **Finish:** Only the selected Execution Channel ends. Other Execution Channel can still send commands and the script execution can finish successfully.

To achieve parallel execution multiple Execution Channels need to be created with multiple **Create API Channel** script commands using a unique name for each Execution Channel.

16.2 Functions

The functionality available in the interfaces shown in the diagram is:
vControl provides functions to start/close/login to vControl.

IRuntimecontroller allows to list, start, stop and interact with methods.
Both of these also send events.

ICedInfo allows to view and react to errors&dialogs.

IExecutionChannel allows to open parallel interfaces for interaction while a method is running.

RuntimeController
<ul style="list-style-type: none"> + CloseMethod(): void + CloseQueryAtStartupDialog(bod): void + GetAllMaintenanceMethods(): string[] + GetAllRunnableMethods(): string[] + GetAllUserNames(): string[] + GetCurrentUser(): string + GetFluentStatus(): StateMachineStates + GetProgress(): int + GetprogressInitialization(): int + GetQueryVariableNames(): string[] + GetStateAsString(StateMachineStates): string + GetVariableNames(): string[] + GetVariableValue(string): string + InitializeInstrument(WorkspaceSelection, string): void + InteriorLight(bool): void + IsReady(): bool + LoginUser(string, string): bool + PauseRun(): void + PrepareMethod(string): void + ResolveExpression(string): string + ResumeRun(): void + RunMethod(): void + SetVariableValue(string, string): void + StopMethod(): void + ValidateUser(string, string): bool <<event>> + ChannelCloses(): ChannelChange + ChannelOpens(): ChannelChange + CommonErrorDialog(): CEDNotification + EnterReadyMode(): Acation + Error(): Notification + ModeChanged(): ModeSwitch + ProgressChanged(): Progress + DeckCheckDiscrepancyDetected(): DeckCheckDiscrepancyEventArgs



NOTICE

SetVariableValue(string,string): void

Valid only for variables of scope Run. They can be set at runtime only.

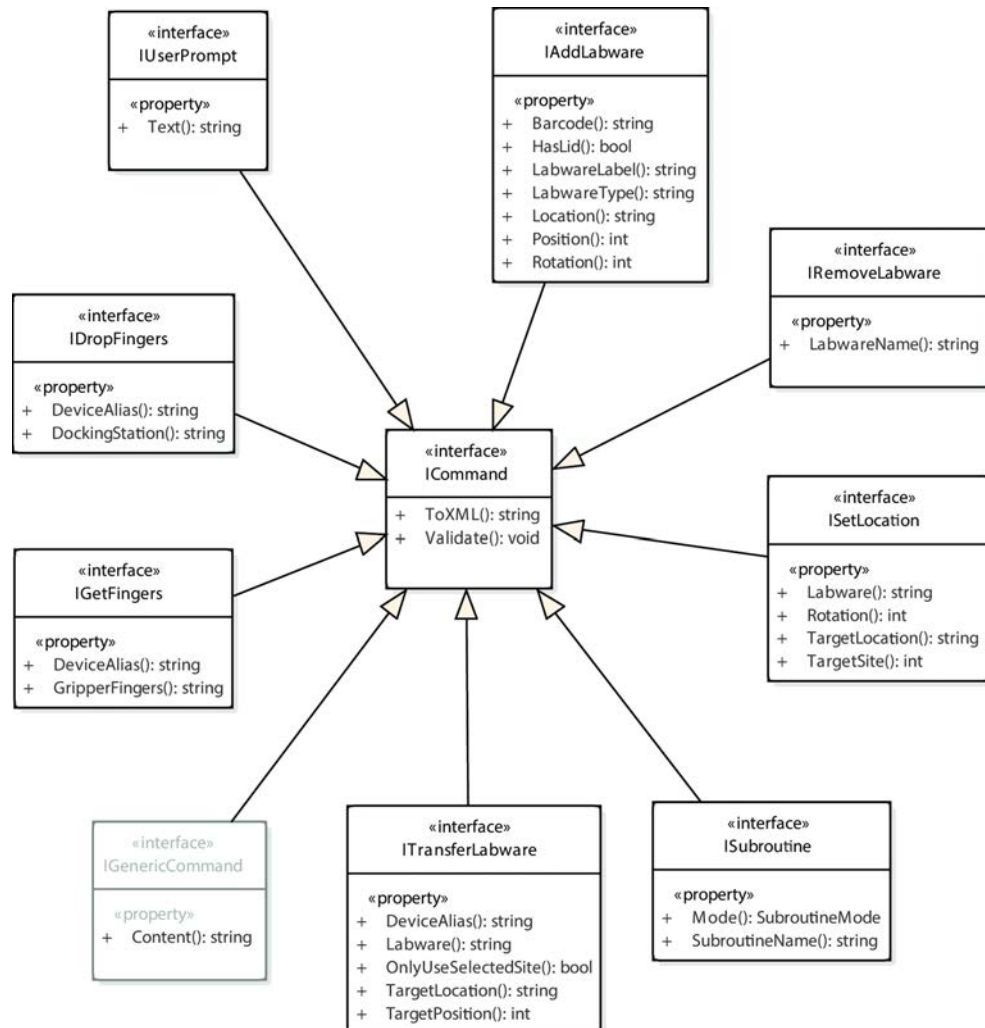


Fig. 181: Commands



The **IGenericCommand** is meant as a supplement for actions that need to be used but were not foreseen in the API (e.g., to set temperatures of devices controlled through third-party drivers or similar). Since the command is very generic is not possible to test in detail what will and will not work. The XML input string is not documented and can change between software versions—i.e., a generic command that worked with a specific vControl version may not work with a more current one.

The function **ReadyToConnect** is deprecated and should not be used anymore. Please use **RuntimeIsAvailable** to determine when to request a runtime. See example "[Start vControl and use API](#)" [365] for details.

16.3 Examples

16.3.1 Start vControl and use API

The API can be used to start vControl. During the first seconds of the startup process the API cannot do anything with vControl. As soon as vControl has loaded and prepared enough data, the following event will be triggered:

`RuntimeIsAvailable`.

When this event is triggered the API user can call all available methods of the **RuntimeController**. The **RuntimeController** provides access to the main functionalities of the API to control a running vControl process. Some of the functions depend on certain states (e.g., `Prepare Method`—which only works in **EditMode**).

C#

```
private static ManualResetEventSlim
_waitForRuntimeIsReady=new ManualResetEventSlim(false);
public static void StartvControl()
{
    vControl process = new vControl();
    bool alreadyStarted = process.IsRunning();
    process.RuntimeIsAvailable +=
_process_RuntimeIsAvailable;
    process.StartOrAttach();

    if( !alreadyStarted )
    {
        _waitForRuntimeIsReady.Wait();
    }
    RuntimeController runtime =
(RuntimeController)process.GetRuntime();
}

private static void _process_RuntimeIsAvailable()
{
    _waitForRuntimeIsReady.Set();
}
```

VBA

```
Dim WithEvents vision_x As vControl

Dim WithEvents run As RuntimeController

Public Sub StartvControl()
    Set vision_x = New vControl
    vision_x.StartOrAttach
End Sub

' Event Handler
Public Sub vision_x_RuntimeIsAvailable()
    Set run = vision_x.GetRuntime
End Sub
```

16.3.2 Prepare a Method Run

Prerequisites for preparing a method for execution with API:

- RuntimeController instance is available
- Otherwise the example code will wait until EditMode is reached.

```
C# private static readonly ManualResetEventSlim _waitForEditMode
    = new ManualResetEventSlim( false );

RuntimeController runtime =
    (RuntimeController)_process.GetRuntime();
runtime.ModeChanged += Runtime_ModeChanged;
if (_runtime.GetVeyaStatus() != StateMachineStates.EditMode)
{
    _waitForEditMode.Wait();
}
runtime.PrepareMethod("Method to prepare");

private static void Runtime_ModeChanged( StateMachineStates
old, StateMachineStates current )
{
    if (current == StateMachineStates.EditMode)
    {
        _waitForEditMode.Set();
    }
}
```

16.3.3 Get ExecutionChannel and Execute Command on Channel

Prerequisites for passing script commands from the API to vControl:

- A script with a **Create Channel** script command

The API Client needs to attach to following **EventHandler** to get an Instance of **IExecutionChannel**:

```
runtime.ChannelOpens += Runtime_ChannelOpens;
```

As soon as the script run is executing the **Create Channel** script command the event handler will be triggered and the API client is now able to send script commands via the **IExecutionChannel** to vControl:

```
private static void Runtime_ChannelOpens( IExecutionChannel
openedChannel )
{
    openedChannel.ExecuteCommand(new
TransferLabware("labware", "targetLocation", 2, true));
}
```

16.3.4 Asynchronous Subroutines

The execution of asynchronous subroutines or the creation of an API channel from within another API channel is not supported. For multiple API channels, a script with multiple Create Channel script commands must be used.

16.3.5 Resolving Expression from PowerShell command line

When vControl is running, the API can be used to get information from the current run. API does not support expressions for script or worktable specific functions.

PowerShell script

```
$veya = New-Object -COM "Tecan.vControl"
$veya.StartOrAttach()
$runtime=$veya.GetRuntime()
$iteration=$runtime.ResolveExpression("loop_iteration")
$volumeWellOne=$runtime.ResolveExpression("GetAttribute("tar
getLabware","Well[1].Volume")")
$volumeWellTwo=$runtime.ResolveExpression("GetAttribute("tar
getLabware","Well[2].Volume")")
$status = @("Iteration: ", $iteration,"Volume 1: ",
$volumeWellOne,"Volume 2", $volumeWellTwo)
$status | format-table
```

17 Scheduler

The vControlScheduler requires an additional license to activate the features described in this chapter.

17.1 The Process Editor

A Process is a set of connected steps. It can be executed as a part of a Method together with vControl standard Scripts.

17.1.1 Creating a Worktable

For setting up processes it is important to create a worktable beforehand. By drag-and-drop all carriers and labware (movable and static) can be placed on the worktable. It is the same procedure as known from scripting in vControl. Labware that will be moveable during a process such as assay microplates are typically placed in storage carriers. For some of those storage carriers like hotels or carousel towers, vControl opens the Labware Group Editor tab. Drag-and-drop the labware into the list below the graphic, assign a labware group name and select at which carrier sites the labware of the selected group is generated. It is recommended to add all fixed Worktable elements like carriers, devices to the Base Worktable. Elements associated with an individual protocol can be added via the Process Worktable.

17.1.2 Open a Process

Navigate to the Control Bar item **Processes** and open a new process by double-click on **New....** Under **Process properties** a process name can be assigned and a worktable from the drop-down list can be linked to the process.

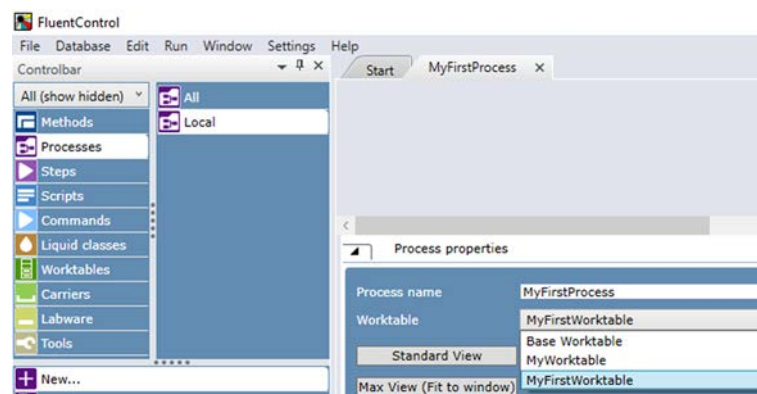


Fig. 182: Open a Process

17.1.3 Process Worktable

It is recommended to add source Labware Groups in the process editor worktable. The added Labware will not be shown in the processing scripts editor worktable, which allows you to use the storage carriers e.g. Hotels for Base Incubation even though it is fully occupied.

17.1.4 Process Steps

For creating a process three different process steps are available: Processing, Source and Destination step.

By drag-and-drop the steps can be added to the process and labware sequences can be created.



Fig. 183: Process Steps

17.1.4.1 Source Step

The Source step is the first step of a labware sequence that defines the labware group which is used in a process sequence. The name of the labware group can be added in the text field of the Source step. It is also possible to use variables in the labware name field. The Source steps can be connected to Processing Steps.

17.1.4.2 Processing Step

A Processing step is a generic node that contains a script (equivalent to vControl Script Editor) when you double-click on it. This script defines which action takes place (pipetting with FCA, Incubation, etc.). The Processing step is set after a Source step or another Processing step.

A descriptive Alias can be given to its text field. A Processing Step by itself is not pre-assigned to a certain driver or resource. The software does automatically figure out which driver's script commands are used within the step script and dynamically change the color of the node according to the assigned driver after the user defined commands in the step. The colors that are assigned to a driver can be changed at **Configure System**.



Fig. 184: Processing Steps

After opening a Processing step the Processing step script name and the expected runtime of this step can be set. Both can be edited in script settings. Changing the expected duration of the processing steps in the corresponding script helps to have a better schedule (refer to section [“Expected Duration of a Processing Step”](#) [▶ 375]). Fresh context-check errors within a script can be indicated at Step level too (refer to [“Context Check”](#) [▶ 379]).

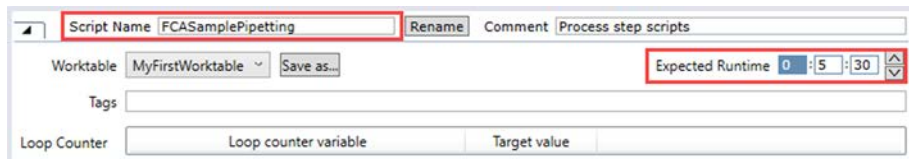



Fig. 185: Sample Transfer

Scripts from Processing Steps are not accessible via Controlbar category Scripts. However, the script names are used at Timing Constraints and Reports. Hence giving descriptive script names is useful.



Processing Step Scripts cannot be saved via the File menu ‘Save’ or ‘Save as...’ options. The Save button below the Script Editor must be used instead.

By drag-and-drop moveable labware has to be set on the worktable (directly onto a Nest of that location) which the system shall automatically transfer to a certain location. The commands can be used as in vControl standard Scripts to program the pipetting or other actions of the Processing step script using the previously added labware.

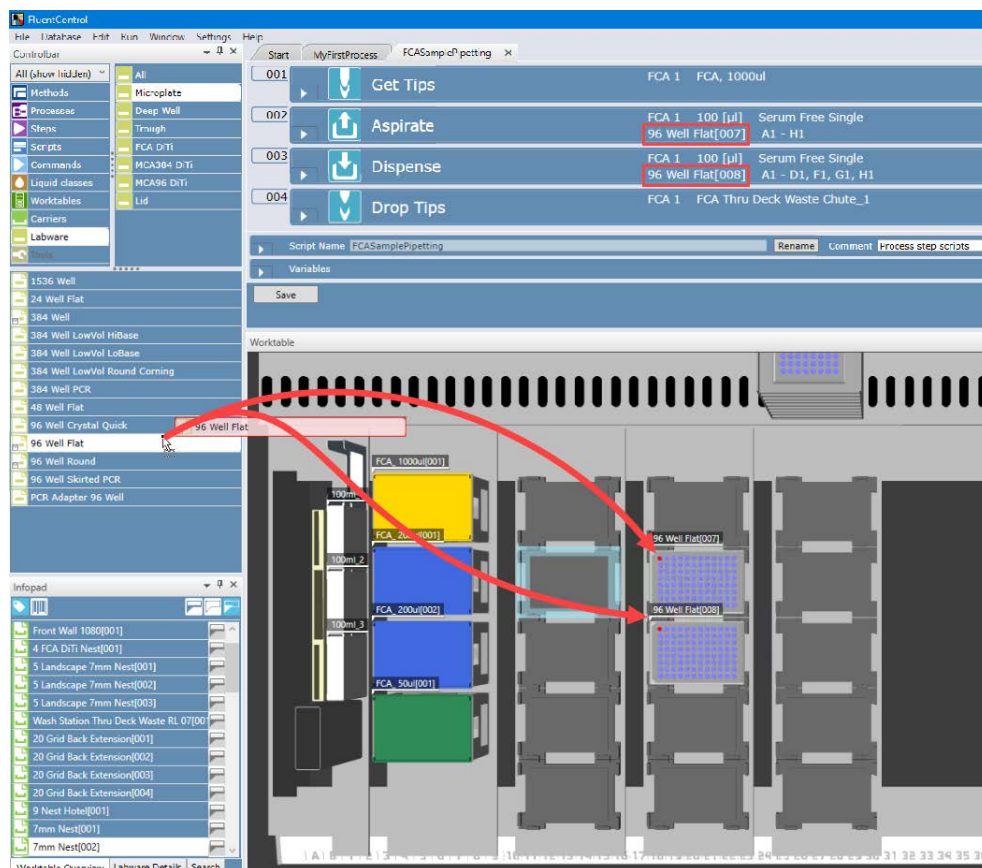


Fig. 186: Moveable Labware

It is possible to add one or more dynamic labware to a processing step. Depending on the complexity of a given process more than five dynamic labware per step can lead to reduced performance. It is also possible to define script variables in the Processing step script, which is similar to the vControl standard Script. Once you have finished the Processing step script, save it, and return to Process.

17.1.4.3 Destination Step

The destination step is an optional final step of a labware sequence and defines the destination of the labware. If no destination step is added the default is to return the labware to its base location.

17.1.5 Process Workflow Definition

The software automatically detects the differences between the *Base Worktable* of the *Process* and the script specific worktable and displays additional labware from the script as labware parameters at the bottom part of the Processing steps.

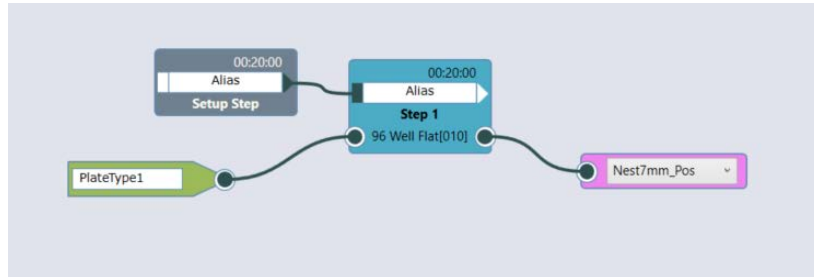


Fig. 187: Connected Steps

Intended use of Transfer Steps and correct use of Destination Step

Only one plate handling command must be used in a Transfer Step. The intended and optional use of customized Transfer Steps is to define additional:

- Commands to move disposable tip labware

Each labware defined in a Source Step in the Process will be brought back to its <base> after all Process Steps have been executed. There is no need to add a Destination Step with location <base>. If a plate shall be brought to any other place than <base>, a Destination Step defining the desired location must be added as the final Process Step of that labware (pink Step in figure below).

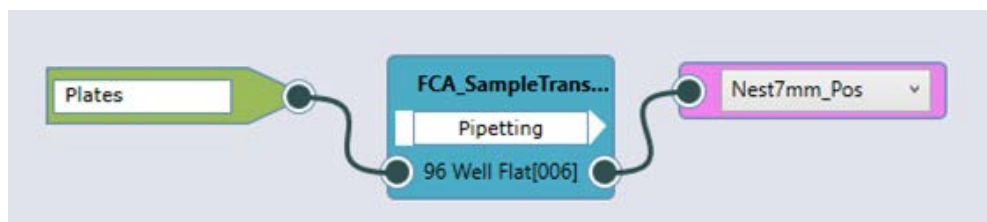


Fig. 188: Connected Steps

If process steps shall be used more than once in the same way as defined, the user has the possibility to duplicate single steps by using copy and paste via context menu or keyboard shortcuts. In the same way the steps can also be deleted. To define a process sequence the labware from Source steps is to be connected to the defined labware of a *Processing* step to give them an execution order. All the process sequences define the later workflow for the process. This means at runtime a labware from this labware group will be assigned to the connected labware parameter of a Processing step and the RGA automatically transfers that labware to the location used on the worktable of the *Processing* step. If necessary, a default transfer step can be customized to add or remove commands to adjust the labware transfer to the actual needs. To do that right click the connection to be customized and select the “*Customize selected connection*” option. A new tab will be opened with the commands of the transfer script and a selection of available commands like for scripts will be shown. A customized transfer step will be indicated by a blue square with three white lines.

Do not change the predefined variables of the “*Transfer Labware*” script command when customizing a transfer step, otherwise there can be problems in combination with a destination step

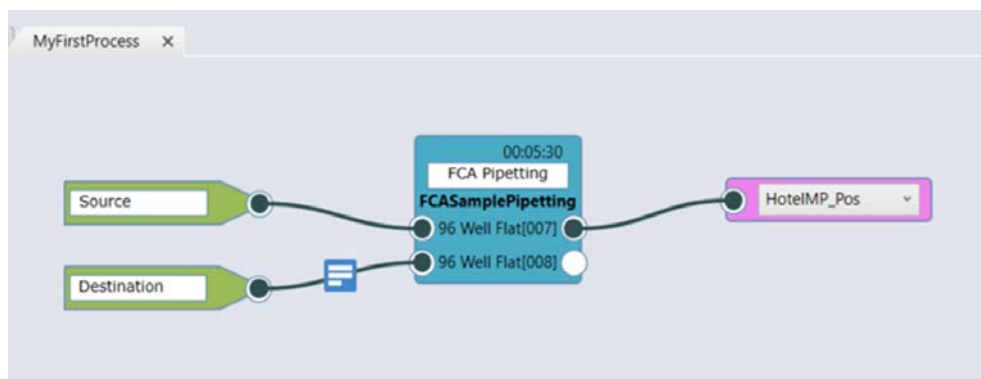


Fig. 189: Connected Steps

The steps can only be connected when the same labware type is used. Note that a step can have different labware types assigned (e.g., '96 Well Flat' and '384 Well'). The connections can also be deleted via context menu or using the Delete key of your keyboard.

The Process Editor can be zoomed in and out diagonally using Ctrl + mouse wheel.

17.1.6 Process Properties

17.1.6.1 Expected Duration of a Processing Step

The expected duration of a Processing step can be set manually in a section of the script.

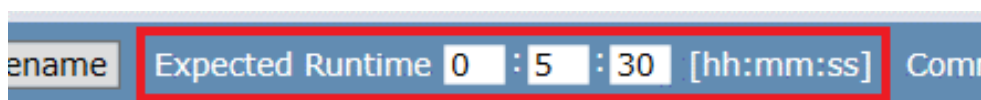


Fig. 190: Expected Runtime

The expected duration time will also be displayed in the tool tip of the Steps in the Process editor tab.

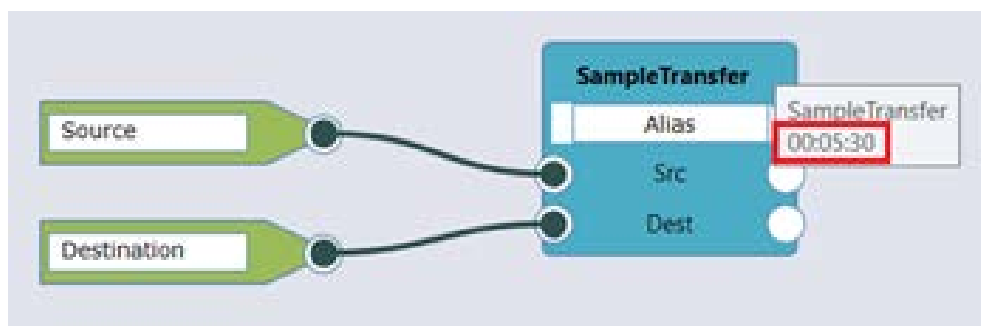


Fig. 191: Connected Steps

The expected runtime of Processing step script shall be as accurate as possible to obtain the best possible schedule. However please consider the longest possible cases one should use the time for the highest number of samples (e.g., when pipetting a variable number of samples). Or you may need to account a bit of extra

time if pipetting is expected to be interrupted at runtime by events such as automated tip handling (refer to section [“Automated Tip Handling” \[385\]](#)) to exchange disposable tip boxes.

If processing of the pipetting step at runtime takes longer than the expected duration, the scheduler dynamically re-schedules and assigns more time to the Processing step in 2 minute increments.



Such a 2-minute time extension can occur multiple times. Keep in mind that future steps in the run may be affected when re-scheduling or it could lead to a violation of a Timing Constraint (refer to [“Timing Constraints” \[377\]](#)).

17.1.6.2 Expected Transfer Step Duration

The expected transfer step duration is set to 15 s by default but can also be changed (refer to [“Expected Transfer Duration” \[383\]](#)).

17.1.7 Variables

It is also possible to declare variables in the Process Editor for using them in the created process. The user can set different scopes.

17.1.7.1 Scope of Variables

The Scope defines where a variable is valid and which items can set or get the value of a variable. The variable scope depends on if you plan to exchange a variable value with other Processes, Iterations or Scripts in the same Run. Do not mix up variables by having multiple variables with the same variable name but having different scopes!

Run: Values of a variables with scope Run are valid and accessible to all Processes, Iterations and Scripts involved in that Run. Please refer to 8.4.1 Variables for more information on persisting variables with scope Run.

Process: Values of a variables with scope Process are valid and accessible to all iterations of a Process during the run but it won't be accessible to other Processes in the run.

Iteration: Values of a variables with scope Iteration are only valid and accessible for a specific Process iteration. Each Process iteration gets its own instance of this variable on which it could work on and reset itself to the default setting for the next iteration.

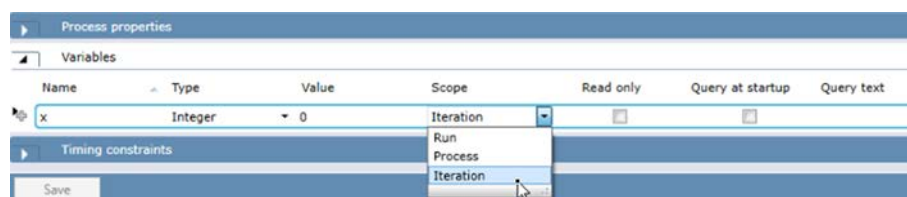


Fig. 192: Process editor tab

Script: Values of variables with Scope Script are only valid and accessible within the Script of a Processing Step.



Scope Script can only be declared when opening the script of a Processing Step.

17.1.7.2 ProcessIteration() function

vControl contains a function called ProcessIteration() which returns the current iteration number of this Process. Imagine it being similar like a loop counter but counting Process iterations instead. When you assign the result of this function to a variable you can use it for programming.

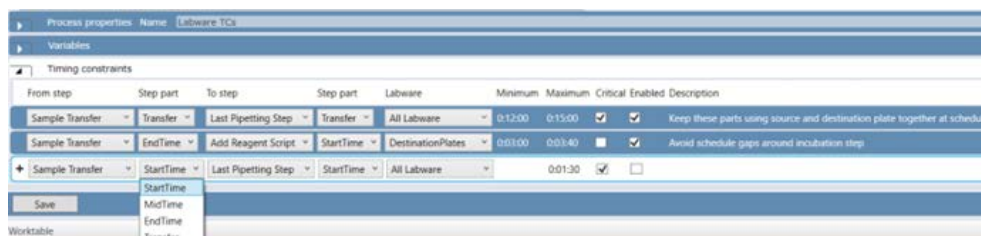
17.1.8 Timing Constraints

Timing Constraints can be set between Process steps, influence the way the schedule of the run is planned and assist you creating a schedule the way you want it to be. Of course, timing constraints will in turn reduce the flexibility of the scheduler in creating a schedule.

They can be defined at a section at the bottom of the Process tab. Select the Step [script name] – from step and to step – and the Step part [StartTime; MidTime; EndTime; Transfer] involved in the time definition. Enter a minimum and maximum duration time and a description.

Note that timing constraints can only be created between process steps which are connected. Either by labware connections or labware-free connections.

When a step is selected in one of the drop-down boxes the other drop-down lists are updated to only show valid entries.



Labware

By default a timing constraint is applied to all labware common to the two selected end points / process nodes. It is also possible to restrict timing constraints to a specific labware. Then the timing constraint only applies to the labware selected in the **Labware** column.

The drop-down list of the Labware column list all labware common to the selected steps as well as a **All Labware** entry.

Selecting a single labware is useful:

- When you want to affect the loading or unloading of a specific labware, e.g. sample plate.
- When you want to limit the reporting of timing constraint violations to specific labware.

Non-critical timing constraints

By default timing constraints are critical, i.e. a violation of a timing constraint will show an error in the Trace View and the run will finish with warnings.

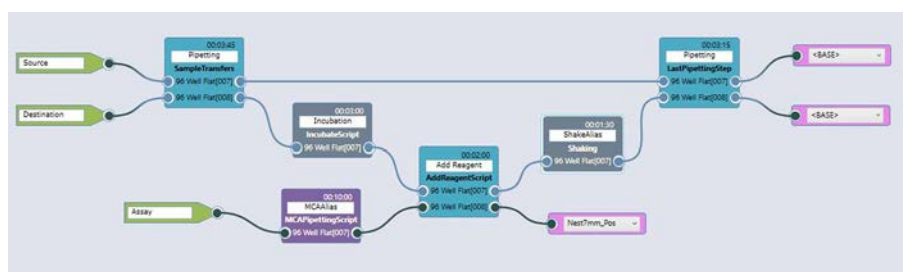
Often a timing constraint is introduced to improve the sequence of the steps or limit on-deck-time of important labware. For these cases a timing constraint can be marked as non-critical in the **Critical** column. If a non-critical timing constraint is

violated during run, only an informational entry is shown in the Trace View. Please note that non-critical timing constraints are not reported in the timing constraints report file.

Enable

In order to quickly test the effect of timing constraints on the schedule, individual timing constraints can be disabled via the **Enabled** column. Please note that disabled timing constraints are not used for scheduling and are not monitored or reported.

Connections with Timing Constraints are highlighted. Connections without Timing Constraints are shown in black. If Step part 'Transfer' is part of the timing constraint, the connected line is colored with a color gradient.



Example

Use of a Timing Constraint

Avoid gaps in the schedule around an incubation since another reagent shall be added after a certain amount of time.



If timers are used to keep the schedule gap closed during an incubation add 10 seconds to the expected duration of the step for step preparation and closing of the step to avoid step delays—e.g.:

A script only containing a Timer and a Wait command that is set to 30 minutes -> use 30 minutes and 10 seconds as the expected duration of that step.

17.1.9 Gantt Chart

During the design time the Gantt chart visualizes the created schedule of the process and provides information about the used labware in the process and the allocated resources. The expected run- and end-time is also displayed, and the user is informed about when a schedule is being calculated or if no schedule is available.

Zooming of the Gantt Chart (not supported by TouchTools)

Mouse Wheel: Scroll Up & Down

[Ctrl] + Mouse Wheel: Extend/Contract X-axis

[Shift] + Mouse Wheel: Zoom, Extend/Contracted X- and Y-axis simultaneously

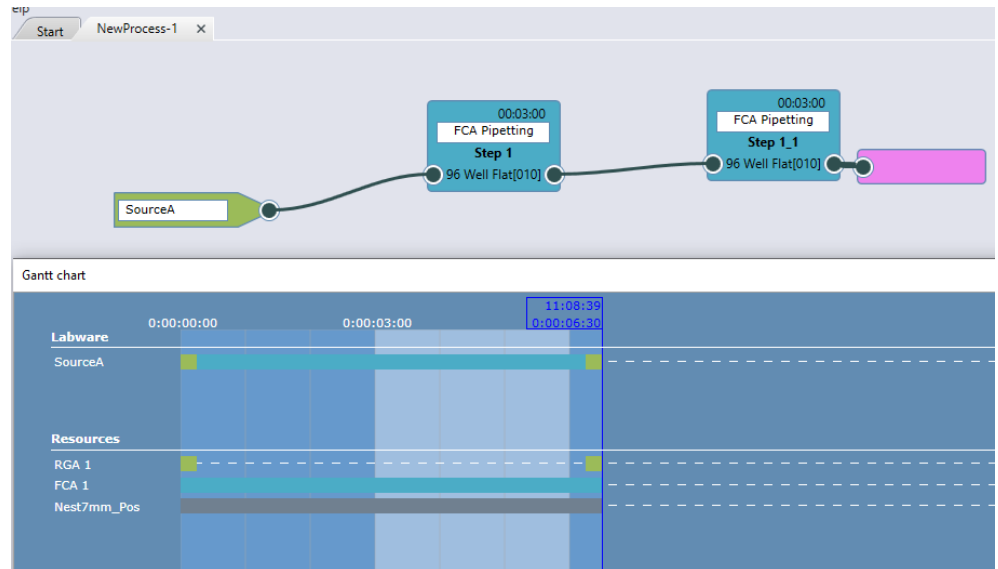


Fig. 193: Gantt chart

For Gantt chart description during run time refer to section Gantt Chart.

17.1.10 Menu Options for Processes

17.1.10.1 Import and Export of Processes

Importing and exporting of processes is possible and works similar to the Import/Export functionality known in vControl for methods and standard scripts. If the user is doing a full export, processes are included. For complex scheduling scenarios a lot of files are needed. It can easily take more than 10 minutes to import such a case.

17.1.10.2 File menu

The user has the option to close an open process when using **Close** and also save an open process under a different name when using the **Save as**.

17.1.10.3 Edit menu

The user has the option to copy, paste and delete processing steps in the opened process.

17.1.11 Context Check

A meaningful context check error text is displayed in the Infopad when something inconsistent is preventing the schedule creation. The process tab is marked with a red flag when a context check error occurs in the process. When the processing step script is marked with a red flag then a context check error exists in this script.

17.2 Define Methods Including Processes

17.2.1 Setting up a Method

When setting up a Method one or multiple processes can be combined into one batch in order to be scheduled in parallel. It is possible to use several script batches in a method.

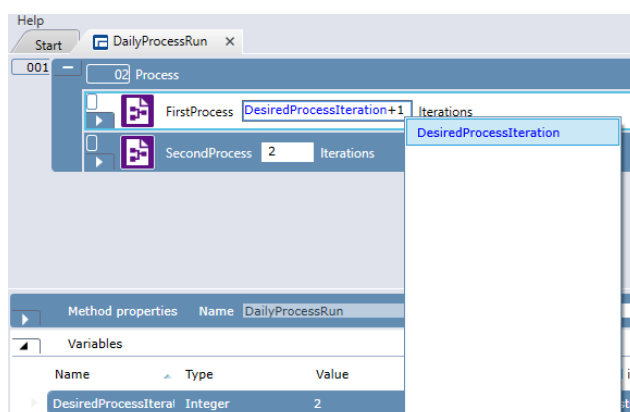


Fig. 194: Setting up a Method

The number of iterations can be defined as an integer, variable with scope Run or expression. If a variable or expression is used, the number of iterations that shall be executed is calculated just before the batch is started.

If the calculated value is not an integer, the default value '-1' will be used.

Expressions Floor() and Round() may be used to ensure that an integer value is used. The Trace View and Audit Trail record the number of iterations started for each batch in the method.

17.2.2 Assign Labware by Barcode File

The **Assign labware by barcode file** option allows you to run processes using specific labware items which are identified by their barcodes instead of using the next available labware. Each process batch item in the **Method Editor** has a small arrow, this can be expanded to enable the option by checking the Checkbox. If you choose to use **Assign labware by barcode file** you should provide either a .txt or .csv file. The file should have the following structure:

```
MyLabware001,MyLabware002,Sample1,Reagent1
ABB123,BAB321,DCC123,AAA123,BBA321
DED123,BAB321,BCD123,BBB123,CDA321 ...
```

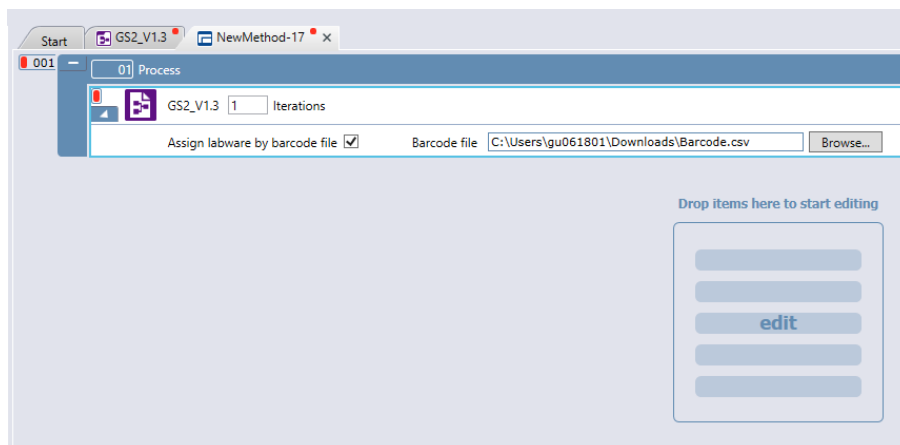



Fig. 195: Process

The first line should contain the list of the source labware group names used in the process Source Steps. The following lines contain the barcodes of the source labware to be used. Each line should include the same number of barcodes as the number of the source labware names in the first line. Each line is used for one process iteration. Empty lines in the file are ignored. You can use semicolons “;” or pipe characters “|” as the separator character instead of commas. The barcodes must not include any of these three separator characters.

The **Assign labware by barcode file** field does not accept variables. If the field is left empty, the operator is prompted to browse for a .csv or .txt file before the run begins. On the touch monitor, it is not possible to browse for the barcode file, this must be done on the PC running vControl.

17.2.3 Touch Tools Settings

In the method's Touch Tools Settings section one can enable **Iteration View** and enter a text that will be shown to the operator at runtime.

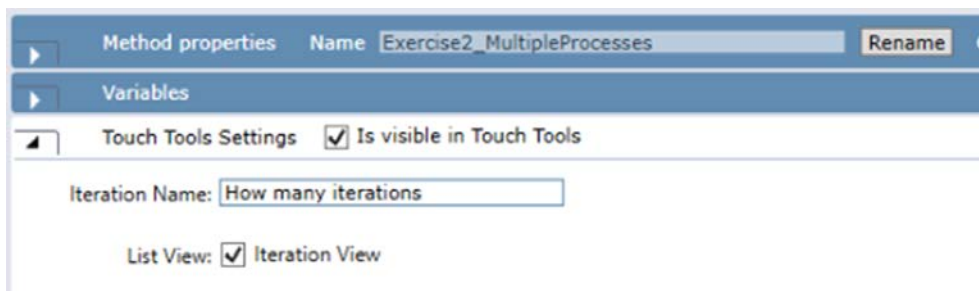


Fig. 196: Touch Tool Setting

Using this method setting allows the operator at Run start to choose how many iterations of each process will be executed in this run.

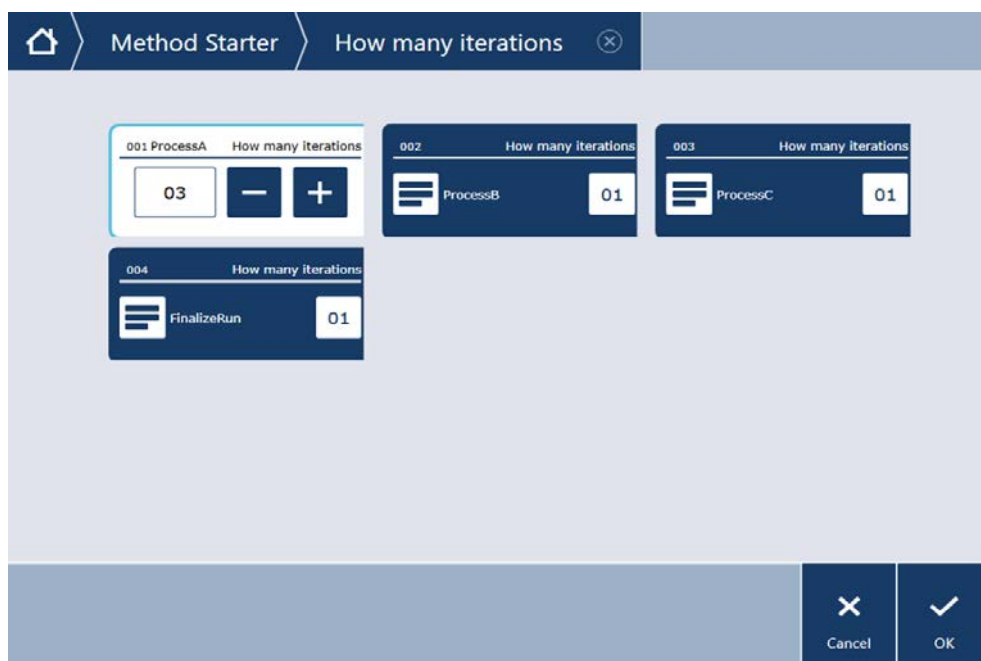


Fig. 197: Touch Tool

17.3 Running a Process

17.4 User Rights

As with standard vControl scripts and methods only the admin user and key operator user have access rights to edit processes. Operators can only run a validated method.

17.5 Scheduler Configuration

Under 'Configure System' the following parameters can be edited.

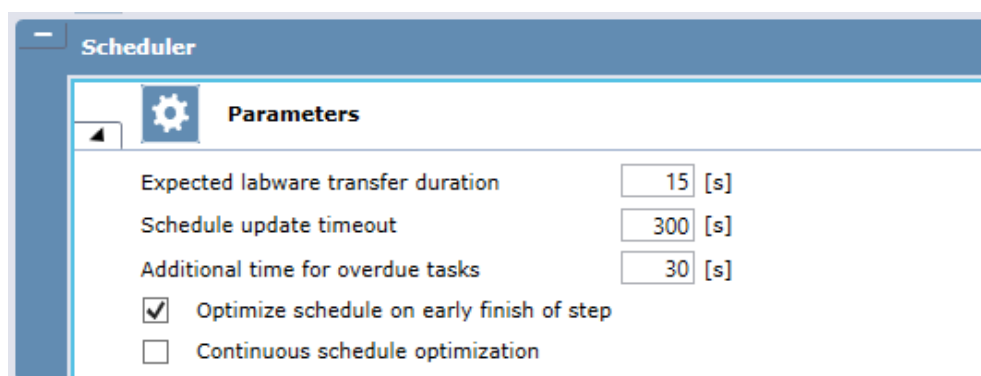


Fig. 198: Scheduler

17.5.1 Expected Transfer Duration

The default expected duration of a transfer step is 15 s. The user has the option to set a new expected duration for transfer steps in **Configure System** under **Scheduler** and **Parameters**.

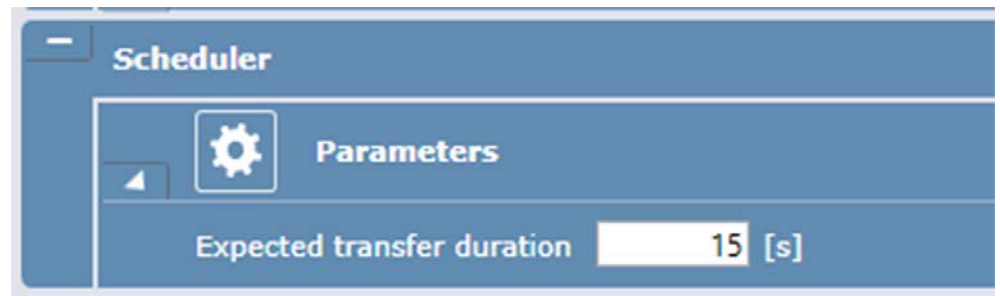


Fig. 199: Expected Transfer Duration

17.5.2 Schedule Update Timeout

The schedule update calculation time has a default timeout of 5 minutes—i.e., if a schedule calculation takes more than 5 minutes, the execution continues with the existing schedule. The user has the option to change the **Schedule update timeout** in the **Configure System** under **Scheduler** and **Parameters**.



17.5.3 Additional Time for Delayed Steps

If a processing step doesn't finish in expected time, additional time for this step to complete is provided. Default is 30 Sec. The user has the option to change the **Additional time for delayed steps** in the **Configure System** under **Schedule** and **Parameters**.

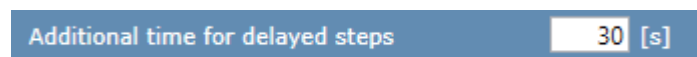


Fig. 200: Additional time for delayed Steps

17.5.4 Optimize Schedule on Early Finish of Step

If a processing step completes earlier than expected the entire schedule will update dynamically (to save time if possible). This setting is enabled by default. Depending on the complexity of the process, this can lead to a deterioration of the schedule. The user has the option to disable the re-scheduling on early finish of steps in the configuration settings (in **Configure System** under **Scheduler** and **Parameters**: disable checkbox **Optimize schedule on early finish of step**). If the re-scheduling is disabled, the system would wait to continue with the following step as it was initially scheduled.

☒ Optimize schedule on early finish of step

17.5.5 Continuous Schedule Optimization

If a processing starts or completes the entire schedule will update dynamically (to save time if possible). This setting is disabled by default. Depending on the complexity of the process, this can lead to a deterioration of the schedule. The user has the option to enable the continuous optimization in the configuration settings (in **Configure System** under **Scheduler** and **Parameters**: enable checkbox **Continuous schedule optimization**).

☐ Continuous schedule optimization

17.6 The Labware Group Editor

As already described in section “[The Process Editor](#)” [▶ 370] when using Scheduler functionality there is a difference between static and movable labware. The movable labware is used for automated transfer to certain locations and defines the labware sequences of a process. The movable labware can be set when designing a worktable. Therefore, storage carriers are used (e.g., hotels or carousel towers). With these settings the source for the labware sequences can be defined and used in process creation. For this, vControl opens the Labware Group Editor tab and the user is able to select the required labware type in the Controlbar. In the Processing step Scripts this movable labware can be set on different worktable locations, like nests. In this way, the user tells the software where the previously defined labware from the source or storage location shall be transferred to in order to be processed.

Static labware can also be set when creating the worktable. This usually encompasses non-processable labware like troughs, reagent reservoirs or carrier-like labware. Static means, that this labware is not automatically transferred from a source or a previous location to another defined location. But note, if placing the labware on a storage carrier, like hotels or carousel towers, the labware will be automatically defined as a dynamic labware. Labware which is placed on the worktable (e.g., an Nest location) in the Processing step Script is not a static labware.

17.7 Devices

Using devices for Process execution is generally working the same as known from vControl. At this point there are some workarounds for driver usage and the usage of duplicated drivers.

When using the **Hydroflex washer** during execution the user normally has the option to use variables for the ‘Wash’ command to use a flexible number of strips. Note that so far the usage of variables is not possible in combination with this command, so a fixed number of strips must be used.

When using **shakers** (MIOs or other shakers) and more than one plate in the incubator, the user need to do the following: If one of the plates shall be transferred from the shaking incubator to another location, the device stops shaking. The RGA transfers the plate but so far it is not working that the shaking continues for the rest

of the plates after this transfer. To ensure continue shaking, the user has the option to start it manually or working with 'If' commands (system shall check locations, if there is a plate and start shaking again, if there is one).

Also, device duplication is generally working the same as known from vControl. There is a workaround when duplicating devices: to control more than one device of a device group it is also recommended to work with 'If' commands to have a better control about where the plates has to be brought to and executed in. E.g., when using more than one shaker, to have a better control about which shaker has to stop shaking or continue shaking; when using more than one reader, to have a better control about which reader has to be used and start a method.

17.8 Automated Tip Handling

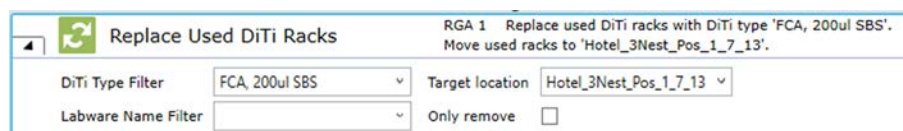
17.8.1 Automated Tip Handling (ADH)

To prevent user interactions during a scheduling process it's possible to program an automated DiTi handling (ADH) option. Disposable tip boxes can be stored in a storage such as a hotel, carousel, or stacker and whenever a tip box on the worktable deck is consumed, the ADH exchanges the empty box with a full box from the storage location. Setting up an automatic DiTi handler works the same way as with conventional vControl scripts.



Fig. 201: On Empty Tray

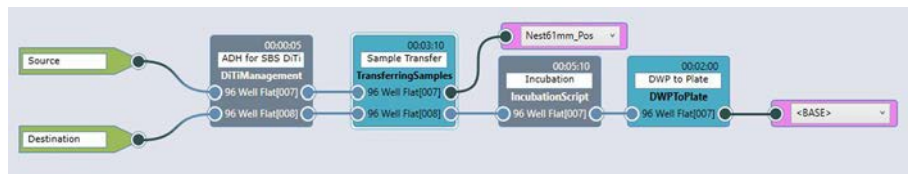
The subroutine can contain any script you like to program, but typically it involves a RGA – Replace Used DiTi Racks command to exchange the tip boxes (or FCA – Remove Empty Tip Spacers in case of Nested FCA Diti & Transfer Tool).



17.8.2 Example Process with ADH for FCA SBS DiTi Boxes

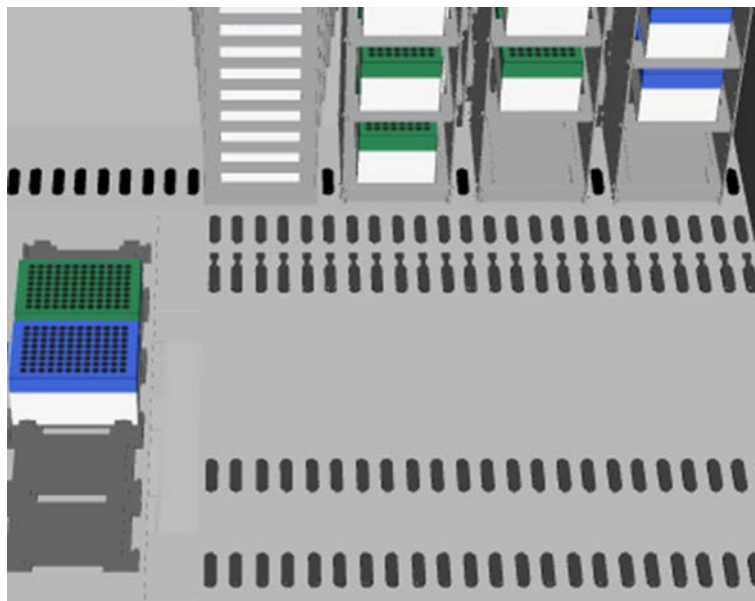
Here is an example script using Automated DiTi Handling. The DiTi Handling is defined in a separate Step at the beginning of the Process and applies then to the entire process. It's not mandatory to have this DiTi management as a separate

step, the 'On Empty Tray' commands could also be included at the start of the first FCA step. It is sufficient to have the 'On Empty Tray' command executed once at the beginning it will apply to other Processing Steps as well.



The DiTi management step contains 'On Empty Tray' commands that trigger the corresponding subroutine to exchange the FCA DiTi boxes.

Please consider in case the first set of tip boxes initially starts on the worktable deck, enough storage space needs to be available for the new boxes at the first tips exchange.



17.9 Validating a Process

For further reducing the risk of potential failure effects, consequences of failures or potential hazards during the execution of processes, pay attention to the following when setting up processes:

- Check that Schedule creation is logical and correct to avoid wrong or partial execution.
- Check after the validating run that the correct process step scripts were executed in the correct order and all timing constraints were met.
- Check that process variables (process-scoped and process-iteration-scoped) are defined correctly to avoid wrong results or any error during execution.

- Check that the correct labware is used to avoid wrong results or any error during execution.
- Consider the usage of controls for all runs to improve process safety.

17.10 Optimizing a Process

For optimizing the execution of a process, the following has to be considered:

- Using minimum incubation across multiple steps: by creating steps that have an expected duration close to the duration of the timing constraint the quality of the resulting overall schedule can be significantly improved.

Bad example:



Good example:



- For creating a better initial schedule, the report can be used to check step duration of the single planned processing steps. ca. 10 s can be added to the actual execution duration of each step. This time can be set as expected runtime of the processing step.
- If a step takes longer than the expected duration, the scheduler dynamically re-schedules and assigns more time to the Processing step in 2 minute increments. This can lead to deterioration of the schedule and bad results. To avoid unnecessary re-scheduling and 2 minutes delay when using a timer or other commands which can be used for an incubation time, please set the expected runtime not as the exact planned and set incubation/delay time. Increase the expected duration by 10 seconds, e.g. the planned and set incubation is 1 h and the set expected runtime is set to 1 h and 10 sec.
- If a step completes earlier than expected the entire schedule will update dynamically (refer to [“Optimize Schedule on Early Finish of Step”](#) [383]). Depending on the complexity of the process, this can also lead to a deterioration of the schedule.

17.11 Support for Cell-Based Assays

For long running assays (multiple days to weeks), the system supports the execution of Tasks. This allows the system to be shut down and restarted for maintenance or other usages during long runs.

Each Task links a Process, along with the labware to use, to a requested start time. This way an assay with incubation times of multiple days can be split into several short Processes. Between these Processes the labware is returned to its base location, for example an incubator. At the requested start time the Process iteration will either be added to an active schedule, or a new schedule will be created.

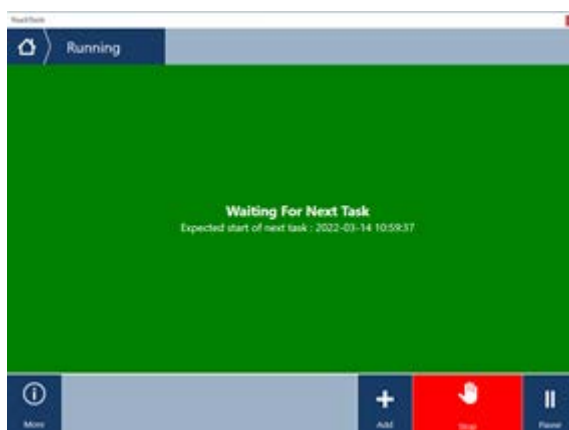
To define Tasks the [“Create Task”](#) [263] and [“Start Task”](#) [263] commands are used. See the commands for details.

When Tasks are added to the system, the system keeps track of them and will try to execute them at the requested start times. If pending Task are present at vControl start up, the software will automatically start a run to execute the Tasks.

The user can cancel this automatic run via a dialog which is shown for 30 seconds. If the start of the run is not aborted within this timespan, the run will be started.

The system may be used as normal while no Tasks are active. vControl will inform the user via the status bar about the time when the system is supposed to start the next Task. If the start time has come, the system will show a dialog as a reminder. Via this dialog the execution can be started or suppressed. If the start is suppressed, the status bar element with the Task information can be clicked to re-open the dialog.

Whenever the Task execution is running and no Tasks are due to be started, the system will stay in Run mode. The vControl main UI as well as Touch Tools will show “Waiting for next task”. The requested start time of the next task is shown in the status bar and on Touch Tools.



While the system is waiting for the next task to become due, it is safe to press the Stop button to end the run. After the run is stopped the system can be used as normal, for example to run maintenance Methods. Be sure not to alter the Runtime worktable unintentionally. Select “Preserve all labware positions from last run” in the Method properties if you are in doubt.

After the system is no longer needed by the user, the status bar can be used to go back to Task execution mode. Click on the element showing the requested start time of the next task in the middle of the status bar.

To allow internal software clean-up processes to run, vControl will perform a daily shut down and restart. The default time for this restart is 23:59. The default can be changed via a .twl file.

For ease of use when defining Processes for cell-based assays, we recommend to add all commonly used labware in the Base Worktable. This way they do not need to be defined in every Process individually and will simplify later changes.

17.11.1 Task Input File (.twl)

To add new Tasks, either use the Create Task and Start Task script commands or add a .twl input file to the Worklist folder: C:\ProgramData\Tecan\VisionX\Worklists. The Worklist folder is constantly scanned for new .twl files while vControl is running.

When a new file is detected, it is processed and removed from the folder. A copy of the file is stored in the Audit Trail as an electronic record.

The structure of the file is defined as follows:

P, Date or relative Day, Time or relative Time, ProcessName, LabwareSource R, Time, Weekdays

The following table specifies these parameters.

Parameter	Valid Inputs	Explanation
Indicator	P or R	"P" indicates that it is a line to add a new process task. "R" indicates that the restart time is set.
Date or relative Day	+0, 2022-03-10, 2022-03-10T15:00, 2022-03-10T15:00Z	+ indicates relative day offset. Absolute Date is given as UCT time or as local time without time zone identifier.
Time or relative Time	+1:00, 15:00	+ indicates relative time starting with the time when the file is read. Absolute time is given as HH:mm
ProcessName	MyProcess	A string of a Process name
LabwareSource	Source(Source[001]), Barcode	A string of the Labware-Group name or a Barcode
Weekdays	1, 2, 3, 4, 5, 6, 7	A list of integers representing the weekdays a restart is allowed to be performed. 1: Monday, 2: Tuesday, etc. This parameter is optional. A restart is allowed on all weekdays when nothing is specified.

Examples:

CellBasedAssay.twl => P, +0, 15:00, TestProcess, NewGroup1(NewGroup1[001])

Restart.twl => R, 23:59, 1, 2, 3, 4, 5

17.12 Not Supported

Please check the following functionalities, which are not supported:

- When using Subroutines or Modules in Processing Step Scripts, make sure that these are not using other resources than the calling Script. vControl will only reserve the resources (arms, devices and Worktable positions) used by the Script of the Processing Step itself. Usage of other resources can lead to unexpected behaviors and abortion of the run.
- More than one process batch per method
- Methods containing processes cannot be recovered in case of an abort of the run
- Worktable manipulation via commands to add or remove labware (e.g., SetLocation) may lead to an unexpected behavior.

18 Utilities

There are two Tecan log viewers available to read log files. Log files are records of the activities the system performed and are mostly used to troubleshoot the system.

18.1 Log Viewer

1. To start the **LogViewer** tool, click **Start > All Programs > Tecan > Tools > LogViewer**.
2. Click **Tools > Filter**.

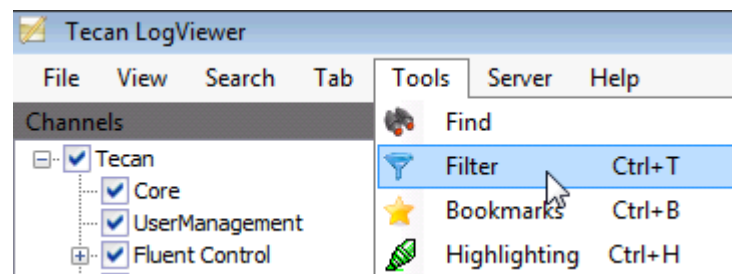


Fig. 202: Tecan LogViewer

3. In the Filter dialog, click **Add**.

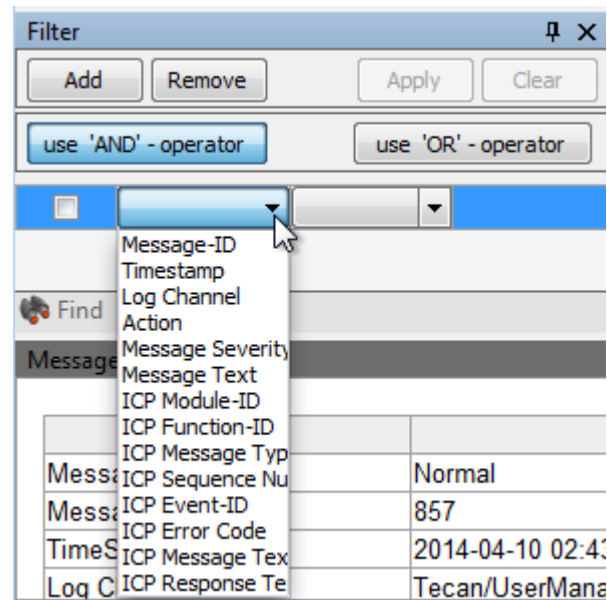


Fig. 203: Select Message Severity.

4. Click the arrow of the first box, and select **Message Severity**.
5. Click the arrow of the second box, and select **=**.
6. Type **Error**.
7. Select the check box, and then, click **Apply**.

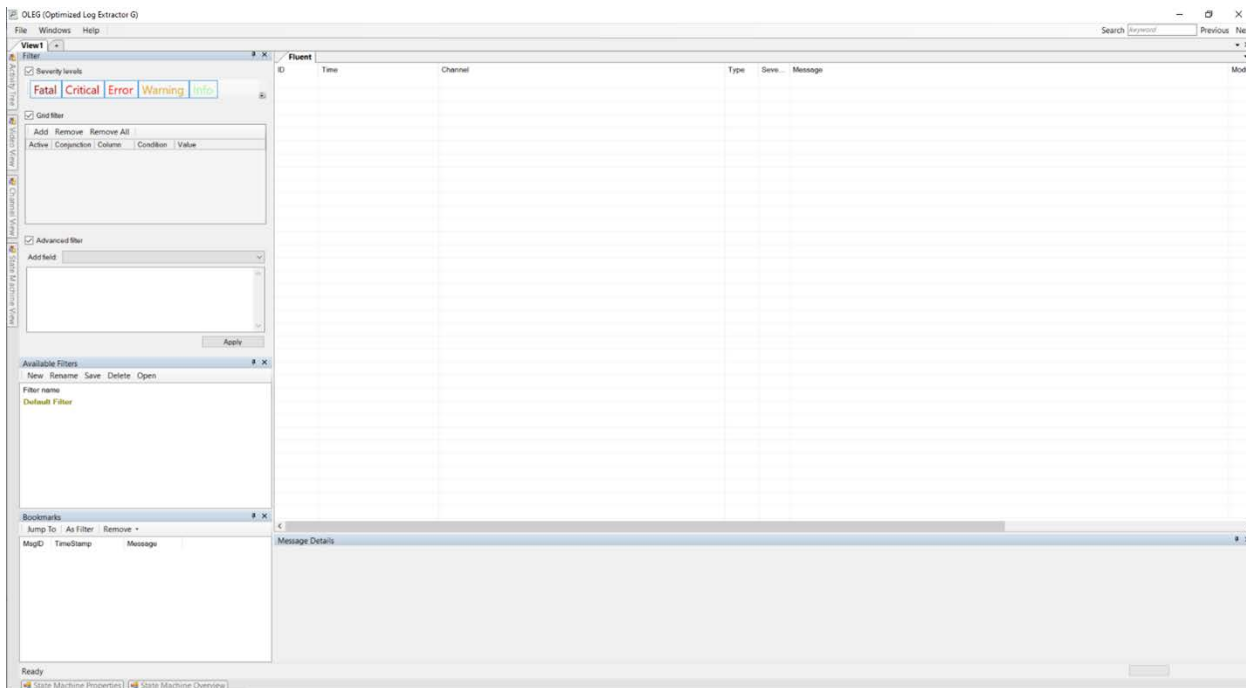
You get a list of all errors for the selected Channels.

vControl supports the following command line parameters

- u <user> with activated user management: pre-select user for login
- w <password> with activated user management: provide password for automatic login
- sim enforces FC start in simulation mode and skips startup questions if no instrument found
- method <methodName> opens specified method
- script <scriptName> opens specified method
- process <processName> opens specified process

18.2 OLEG Log Viewer

1. To start the **OLEG** tool, click **Start > All Programs > Tecan > OLEG**



2. For more information on how to use the OLEG tool consult the OLEG manual accessible through the Help menu within the OLEG application.

19 Customer Support

This chapter explains which files and information Tecan requires to perform a first assessment of an issue.

19.1 Data Collection for Customer Support

When contacting customer support, please provide the following information:

- **Serial number** of the instrument.
- **Detailed description of the problem** that includes:
 - Timestamp of the error
 - Instruction on how to reproduce the error
 - Has daily and weekly maintenance been performed on the system?
 - Video or photo that shows the error
 - LOT number of item for complaints related to consumables
 - etc.
- **Description of preliminary troubleshooting:**
What has been already attempted to resolve the problem?
- **Instrument hardware and software configuration:**
 - Arm order, tip configuration and integrated devices
 - Software and driver versions
- The ZIP file generated with the **Snapshot Tool** as well as **additional files**, if such as:
 - Names of affected scripts or methods and a ZEIA export file including these. [“See section 18.1.2” \[▶ 393\]](#) on how to create an export file.
 - Additional screenshots illustrating the issue.
 - Additional log files. vControl file archiving might be active. To gather archived log files, check the destination folder defined in any active AuditArch archiving actions. For more information on AuditArch, see section 13.4.1

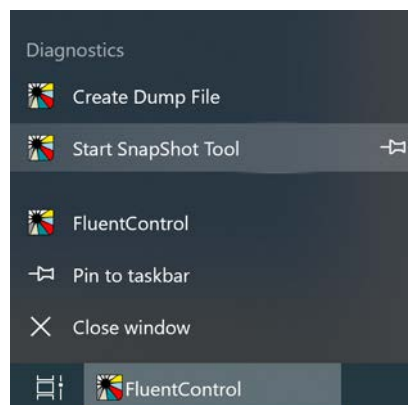
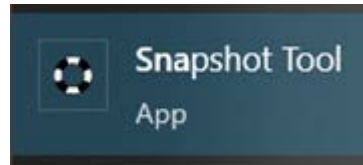
DumpFiles are files that are created when vControl unexpectedly closes. In case vControl freezes and does not respond it is possible to create DumpFiles manually from the jump list, or alternatively from the task manager. To do so, right click vControl in the taskbar and click **Create Dump File**.

19.1.1 SnapShot Tool

The **Snapshot Tool** is used for collecting e.g., *Instrument Files*, *Screenshot*, *Instrument Configuration*, *Firmware Versions*, *Windows Event Logs*, *Journal Files* and *Audit Trail and Sample Tracking Log Files* and logfiles from other devices installed if needed, over certain period of time. The collected information will be stored in a ZIP file for further use and a pop-up window will show where the ZIP file is stored at the end of the collection process.

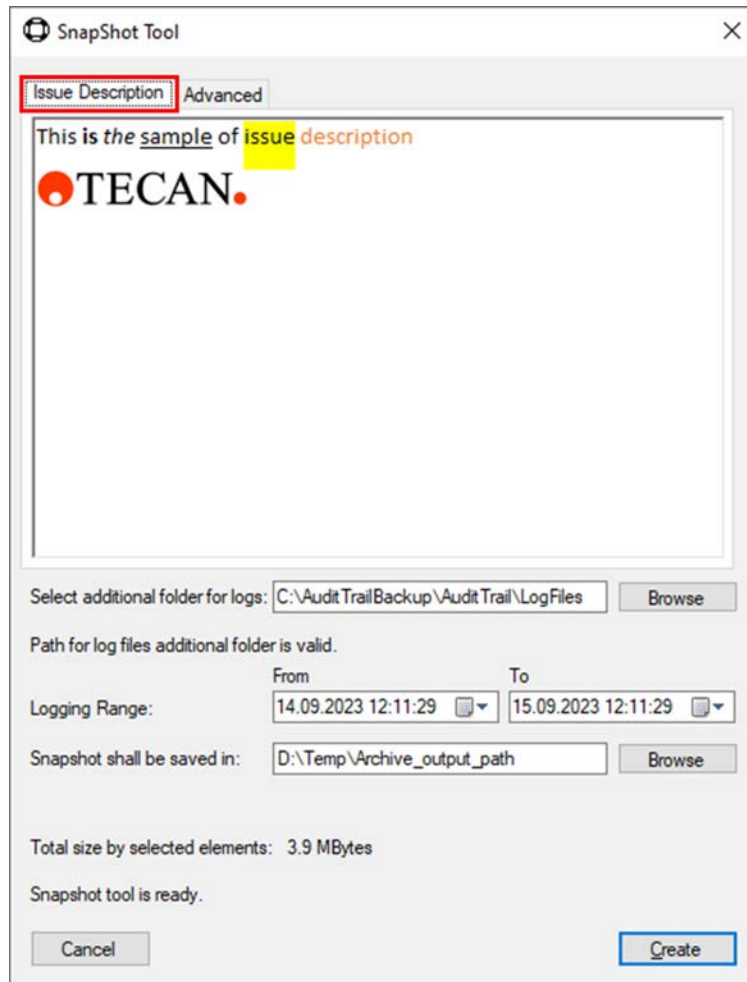
To create a Snapshot, proceed as follows:

1. Open the **SnapShot Tool** from the Windows Start menu, or from the vControl jump list.



2. The **SnapShot Tool** starts with the **Issue Description** tab. Please describe the issue in detail in this window. You can also add drawings, images or further screen shots.
Select additional folder for logs: The user can specify a folder for additional log files which will be included in the ZIP (e.g. DriverFramework log files).
Logging Range: Select the time range within which the issue(s) or error(s) occurred. Narrow down the time frame the issue was observed to limit the amount of data to be sent.
Snapshot shall be saved in: Define the path to save the ZIP-file of the snapshot.

Total size by selected elements: Depicts the total file size of the snapshot.



SnapShot Tool

Issue Description Advanced

This is the sample of issue description

TECAN.

Select additional folder for logs: C:\Audit TrailBackup\Audit Trail\LogFiles Browse

Path for log files additional folder is valid.

Logging Range: From 14.09.2023 12:11:29 To 15.09.2023 12:11:29

Snapshot shall be saved in: D:\Temp\Archive_output_path Browse

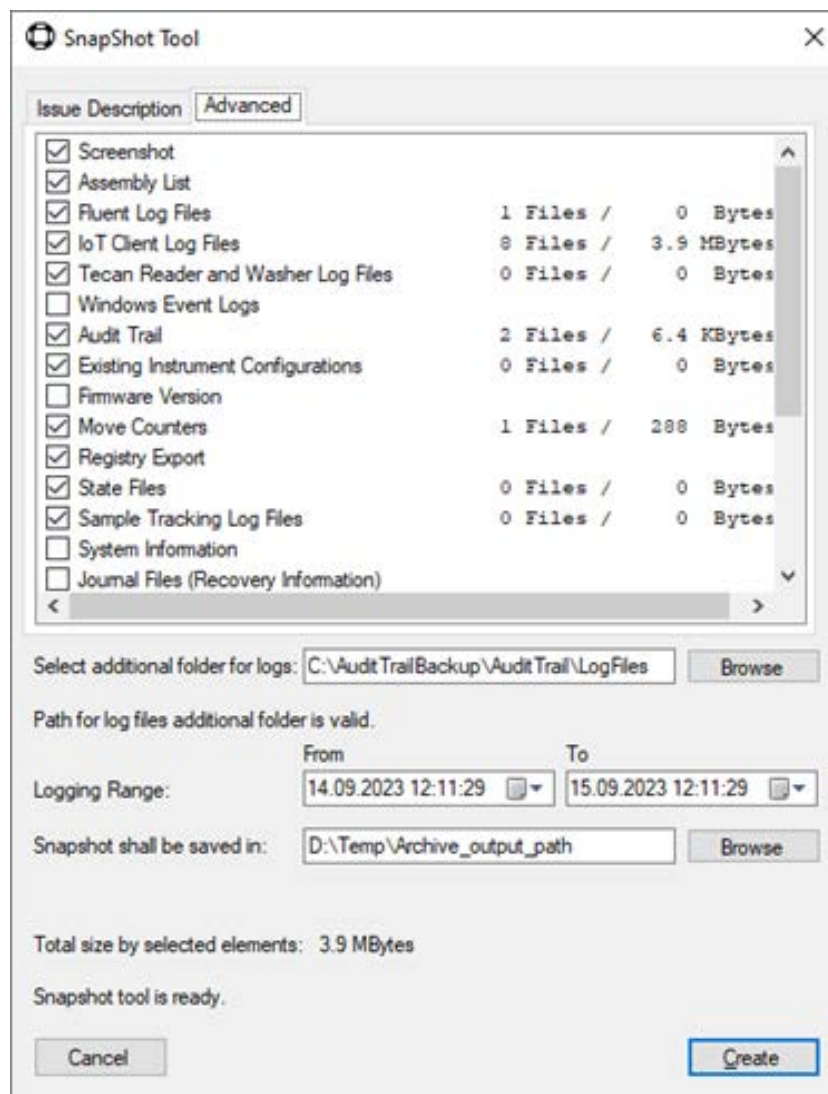
Total size by selected elements: 3.9 MBytes

Snapshot tool is ready.

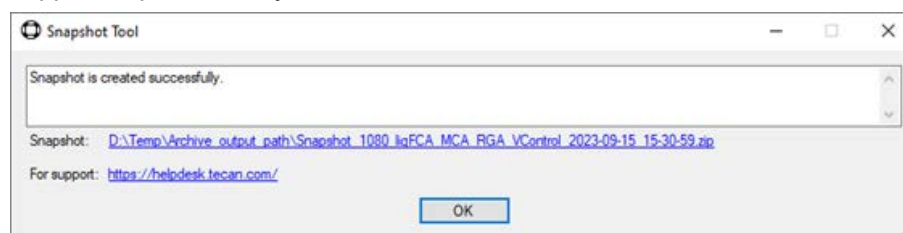
Cancel Create

3. Switch to the **Advanced** tab to select which data shall be collected. Additional data can be added, for example: *Add Sample Tracking Log Files* for Sample Tracking-related issues, add *Journal Files* to collect method recovery data for

issues related to recovery. In case you observed an unexpected shutdown, or a software freeze include *Dump Files*.



- When all data are selected, and a valid path is chosen the “**Snapshot tool is ready**” sign appears.
- Click **Create** to start collecting the data. This procedure might take some minutes. Once data collection is completed a message displaying where the ZIP-file is stored is shown. Click on the **For support** link to create an online support request directly.



Tab. 17: : List of elements that can be included in the snapshot

Screenshot	Takes a screenshot of each monitor, including the touch monitor, after clicking the Create button. Make sure that the vControl user interface is visible on one of the monitors.
Assembly list	Collects the vControl installation files and their version numbers.
Instrument log files	Collects the instrument log files.
IoT Client log files	If the system uses any cloud services, such as Introspect, activate the checkbox.
Tecan Reader and Washer log files	Collects Tecan supplied washer or reader log files.
Windows Event logs	Collects vControl information, if the software has crashed or is frozen.
Audit trail	Collects audit trail information.
Instrument configuration	Collects instrument configuration information. In demo mode, or simulation, no data is collected.
Firmware version	Collects instrument firmware version information. In demo mode, or simulation, no data is collected.
Move counters	Collects move information about arm. In demo mode, or simulation, no data is collected.
Registry Export	Collects all related Tecan entries in the Windows registry.
State Files	Collects the state of each arm at the time of the SnapShot creation.
Sample Tracking log files	Collects Sample Tracking information, if this option is installed on the system.
FRIDA reader log files	Collects FRIDA reader log files, if a FRIDA reader is connected to the system.
System Information	Collects information about the connected computer.
Journal files (recovery information)	Collects Method recovery files. Needed if the issue is connected to Method Recovery. Can be large in size and their collection time-consuming

Database	Collects the content of the vControl database. Only needed upon request by Tecan support.
SVN root	Collects the vControl SVN root, but only upon request by Tecan.
Path Finder Swift Dumps	Collects PathFinder swift dumps. Needed for analysis of pathfinding and collision avoidance issues .
Dump Files	Collects the memory dump files which are automatically created when vControl has crashed.
Driver configurations	Collects the vControl driver settings, but only upon request by Tecan.
User Description	Contains the data the user enters in the <i>User Description field</i> .

Add new entry called "DeckCheck pictures": Collects the pictures, that were taken by the Capture Worktable command of the Deck Check driver in png-format. They are available here: \ProgramData\Tecan\LoggingServer\LogFiles\DeckCheck Pictures.

19.1.2 Export / Import Tool

vControl includes an Export / Import tool for exchanging scripts, liquid classes, carriers and labware definitions with other installations, for technical support or for backup use. The Export / Import tool can be accessed from the main vControl application window and produces files with the *.zeia extension.

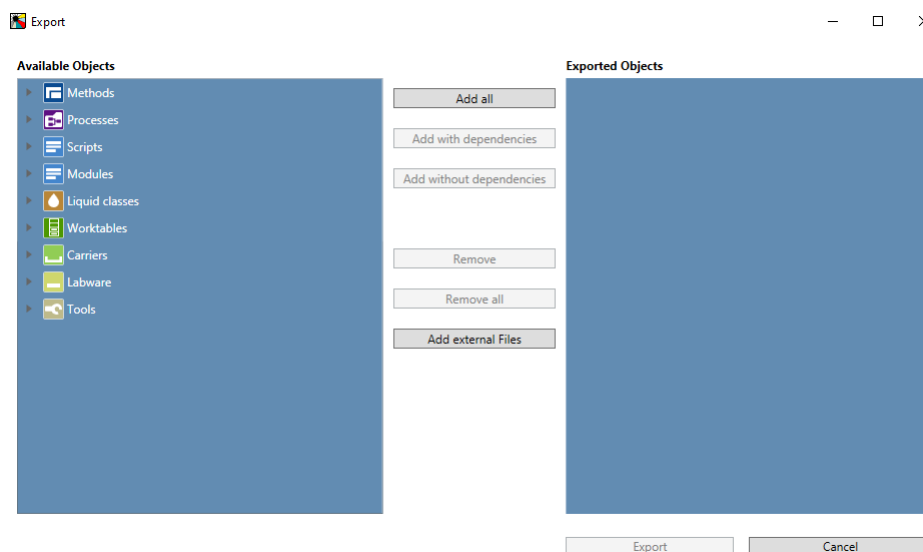
Note: For troubleshooting, use the Export tool to export relevant methods and/or scripts including dependencies as a *.zeia file. To add other external files that are not automatically referenced, such as worklist files, click the “Add external Files” to the exported objects. Make sure all dependencies of a method or script such as carriers, labware, and liquid classes are always included.

1. In the **Database** menu, click **Export...** or **Full Export....**

*The **Export** dialog window opens.*

*The **Available Objects** list on the left is expandable and contains items that are available for export.*

*The **Exported Objects** list on the right contains items that are currently selected for export (empty in the image below).*



2. Select the items to be exported by using one of the following options:

Add all adds all items in the database to the **Exported Objects** list.

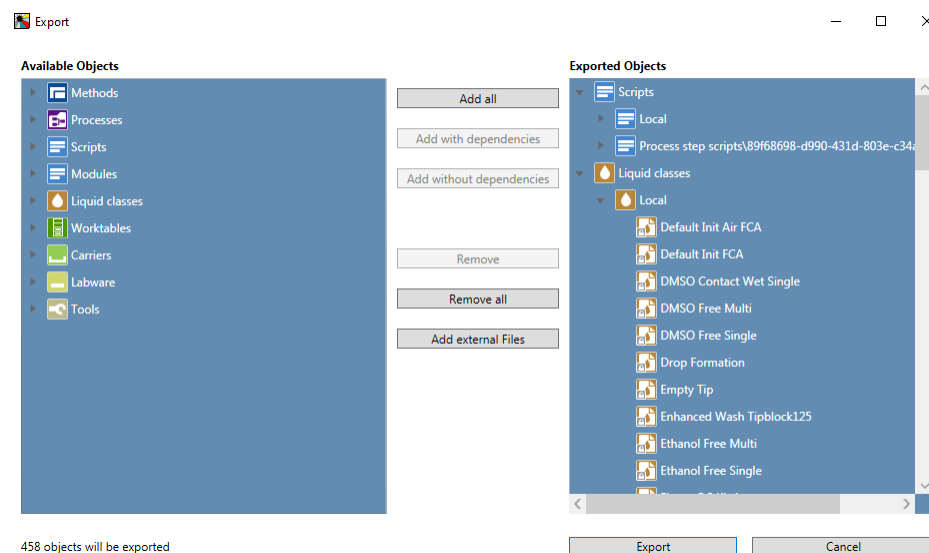
Add with dependencies adds all selected items to the **Exported Objects** list—including their referenced scripts, liquid classes, worktables, carriers, labware and tools.

Add without dependencies adds all selected items to the **Exported Objects** list—without their referenced objects. Note that if any of the referenced items is missing, it may not be possible to import these items successfully to the target system.

3. To include external files in the export (e.g., images, PDFs, etc.), click **Add external Files**.

*In the file browser that opens, select the files and click **Open**.*

*The selected external files will be added to the **Exported Objects** list.*



4. Click **Export** to export all items of the **Exported Objects** list.

19.1.3 Import Mode

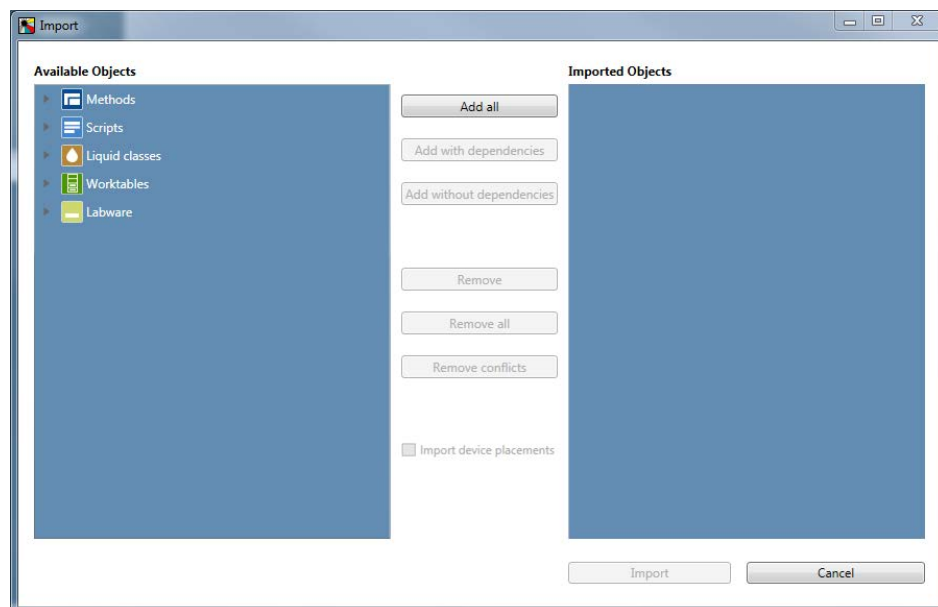
To import a file proceed as follows:

1. On the **Database** menu, click **Import...**,
2. In the file explorer browse for and select a .ZEIA file to be imported.

*The **Import** dialog box opens.*

*The expandable **Available Objects** list contains all items that are available for import. They are arranged in the same way as in the **Controlbar**.*

The **Imported Objects** list contains all items that are currently selected for import.

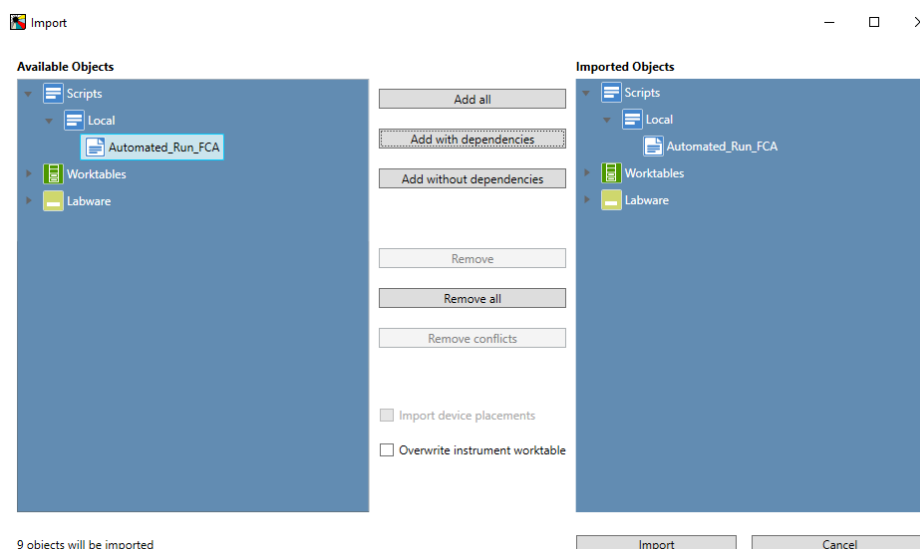


3. Select the items to be imported by using one of the following options:

Click **Add all** to add all items in the *.zeia file to the **Imported Objects** list.

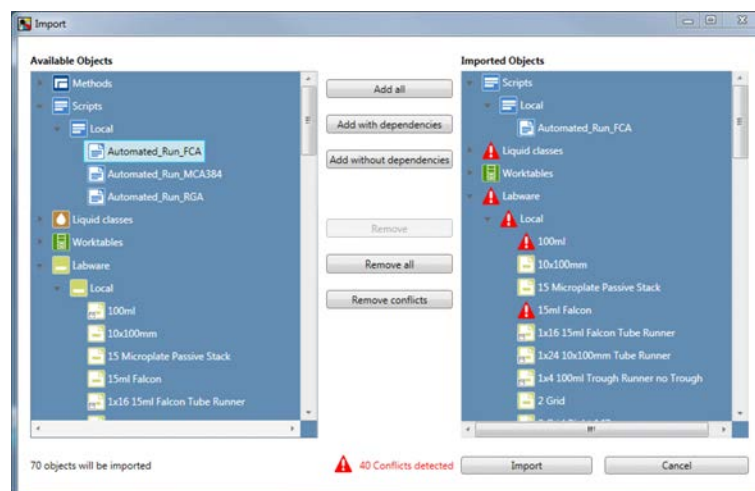
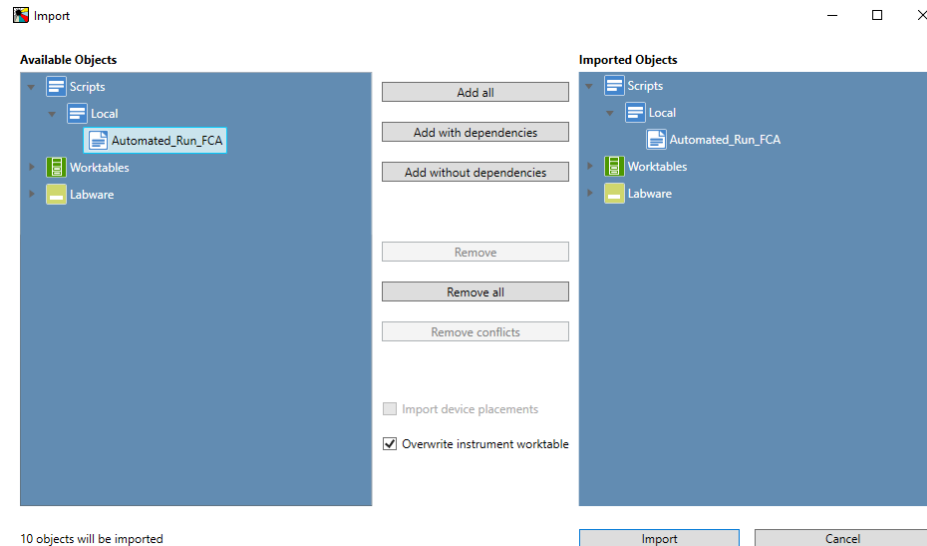
Click **Add with dependencies** to add all selected items to the **Imported Objects** list—including their referenced scripts, carriers, tools, modules, processes, worktables, labware and liquid classes.

Click **Add without dependencies** to add all selected items to the **Imported Objects** list—without their referenced items. Connectors, sites and meshes are part of the import.



If a worktable is imported, it is possible to overwrite the instrument worktable currently present in the application with the one provided in the *.zeia file. To overwrite the instrument worktable, select the checkbox “Overwrite instrument

worktable". For more information on the instrument worktable, please see the chapter **instrument worktable**.



After selecting the items for import, vControl checks for conflicts with the target system's database.

Conflicts occur when items have the same name, type and subfolder but different content. Conflicts are marked with a red alert symbol in the **Import** dialog box. When importing a conflicting item, the item in the target system database will be overwritten.

It is possible to exclude individual items from the **Imported Objects** list, but this may result in context-check errors with imported scripts or methods or missing components on the imported worktable. Carriers, labware or tools that are not present on the target system – but are used by a worktable to be imported – must also be imported.

4. To remove items from the **Imported Objects** list, use one of the following options:

Click **Remove** removes the selected items from the **Imported Objects** list.

Click **Remove all** removes all items from the **Imported Objects** list.

Click **Remove conflicts** removes all conflicting items from the **Imported Objects** list.

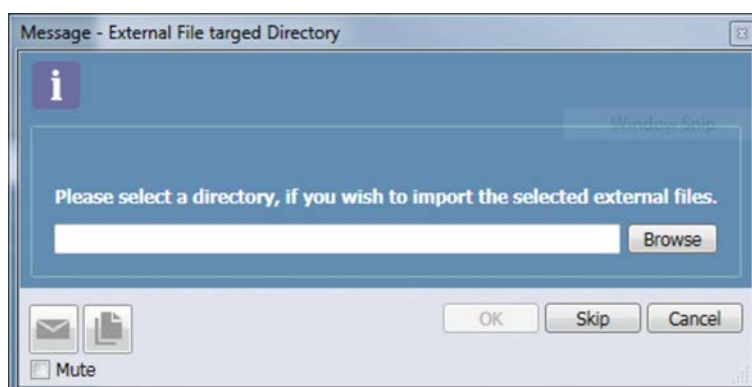
The checkbox **Import device placements** specifies whether it will import placement data for non-grid based device carriers. Overwriting potentially already existing data has only an effect in a simulated environment but not if a physical device is connected, since device placement is stored on the device itself (Te-Control) and will be synchronized automatically.

5. Click **Import** to import all items of the **Imported Objects** list.

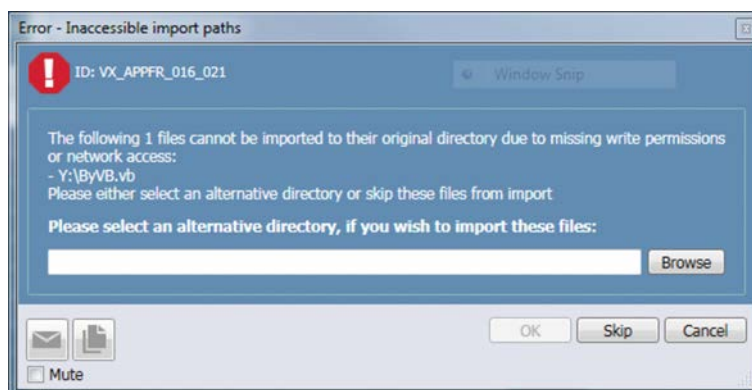
If items to be imported reference scripts, worktables, labware or liquid classes that are not part of the import, a Warning dialog listing the missing referenced items may appear and ask for confirmation.

When importing a .ZEIA with external files, the dialog box below may also appear.

6. To store the external files, click **Browse** and select the desired directory and click **OK**. This directory may be different from the original system directory (e.g., if files are stored on a different drive that does not exist on your own computer). Clicking **Skip** or **Cancel** aborts the import, and the external files will not be stored.



When importing a script with referenced external files (e.g., worklists, VB scripts), the script and the referenced files will be stored in the same folder of the target system. If the target system's folder is inaccessible, the **Inaccessible import paths** dialog box opens. All inaccessible target system folders are grouped by their root directory. The dialog is displayed for each of these root directories.



7. Click **Browse** and select the desired directory on the target system. Then click **OK**.
8. To avoid storing the referenced external files of a root directory, click either **Skip** or **Cancel** to abort the import.

19.2 Contacts

Please contact your local distributor or importer or one of the addresses below.

Also see our homepage on the web: www.tecan.com

Tab. 18: Customer Support contacts

Country/Region	Address	Telephone/Telefax/E-mail	
Asia	Tecan Asia Pte Ltd. 18 Boon Lay Way, #10-106 TradeHub 21 Singapore 609966 Singapore	Phone Fax E-mail	+65 6444 1886 +65 6444 1836 tecan@tecan.com.sg
Australia New Zealand Pacific Islands	Tecan Australia Pty Ltd Unit 2, 475 Blackburn Road Mount Waverly VIC 3149 Australia	Phone Phone Fax E-mail	Toll Free: 1300 808 403 +61 3 9647 4100 +61 3 9647 4199 helpdesk-aus@tecan.com
Austria	Tecan Sales Austria GmbH Untersbergstrasse 1a 5082 Grödig Austria	Phone Fax E-mail	+43 6246 8933 256 +43 6246 72770 helpdesk-at@tecan.com
Belgium	Tecan Benelux B.V.B.A. Mechelen Campus Schaliënhoevedreef 20A 2800 Mechelen Belgium	Phone Fax E-mail	+32 15 42 13 19 +32 15 42 16 12 tecan-be@tecan.com
China	Tecan (Shanghai) Laboratory Equipment Co., Ltd. 1F, T 15-4, #999, Ningqiao Road, Pilot Free Trade Zone, Shanghai, PRC, 201206	Phone E-mail	+86 40 0821 38 88 helpdesk-cn@tecan.com
Denmark	Tecan Denmark, Filial af Tecan Nordic AB, Sverige Lejrvej 29 3500 Værløse Denmark	Phone E-mail	+46 8 7503940 info-dk@tecan.com

Country/Region	Address	Telephone/Telefax/E-mail	
France	Tecan France S.A.S.U Tour Swiss Life 1 bd Marius Vivier Merle F- 69 003 Lyon France	Phone Fax E-mail	+33 4 72 76 04 80 +33 4 72 76 04 99 helpdesk-fr@tecan.com
Germany	Tecan Deutschland GmbH Werner-von-Siemens-Straße 23 74564 Crailsheim Germany	Phone Fax E-mail	+49 1805 8322 633 or +49 1805 TECAN DE +49 7951 9417 92 helpdesk-de@tecan.com
Italy	Tecan Italia, S.r.l. Via Brescia, 39 20063 Cernusco Sul Naviglio (MI) Italy	Phone Fax E-mail	+39 800 11 22 91 +39 (02) 92 72 90 47 helpdesk-it@tecan.com
Japan	Tecan Japan Co., Ltd. Kawasaki Tech Center 580-16, Horikawa-cho, Saiwai-ku Kawasaki, Kanagawa 212-0013 Japan	Phone Fax Phone E-mail	+81 44 556 7311 (Kawasaki) +81 44 556 7312 (Kawasaki) +81(0) 6305 8511 (Osaka) helpdesk-jp@tecan.com
Netherlands	Tecan Benelux B.V.B.A. Industrieweg 30 NL-4283 GZ Giessen Netherlands	Phone Fax E-mail	+31 20 708 4773 +31 183 44 80 67 helpdesk.benelux@tecan.com
Scandinavia	Tecan Nordic AB Sveavägen 159, 1tr SE-113 46 Stockholm Sweden	Phone Fax E-mail	+46 8 750 39 40 +46 8 750 39 56 info@tecan.se
South Korea	Tecan Korea Ltd. 149 Gasan digital 1-ro Geumcheon-gu Seoul South Korea	Phone E-mail	+82-2-818-3301 helpdesk-kr@tecan.com
Spain Portugal	Tecan Ibérica Instrumentación S.L. C/ Lepanto 151 Bajos E-08013 Barcelona Spain	Phone E-mail	+34 93 595 25 31 helpdesk-sp@tecan.com

Country/Region	Address	Telephone/Telefax/E-mail	
Switzerland	Tecan Schweiz AG Seestrasse 103 8708 Männedorf Switzerland	Phone Fax E-mail	+41 44 922 82 82 +41 44 922 89 23 helpdesk-ch@tecan.com
United Kingdom	Tecan UK Ltd. Theale Court 11-13 High Street Theale, Reading, RG7 5AH United Kingdom	Phone Fax E-mail	+44 118 930 0300 +44 118 930 5671 helpdesk-uk@tecan.com
USA	Tecan US, Inc. 9401 Globe Center Drive, Suite 140, Morrisville, NC 27560 USA	Phone Fax Phone E-mail	+1 919 361 5200 +1 919 361 5201 Toll Free in the US: +1 800 TECAN US or +1 800 832 2687 helpdesk-us@tecan.com
USA (Tecan Systems)	Tecan Systems, Inc. 2450 Zanker Road San Jose, CA 95131 USA	Phone Fax E-mail	+1 408 953 3100 Toll Free: +1 800 231 0711 +1 408 953 3101 helpdesk-sy@tecan.com

20 Abbreviations

20.1 Abbreviations Table

This table establishes a list of abbreviations for the Veya:

Tab. 19: Abbreviations table

Abbreviation	Definition
ARP	Air Restriction Pipetting
cLLD	capacitive LLD (pLLD -> pressure LLD)
CV	Coefficient of variance or variation
cXP	cXP Diluter
DiTi	Disposable Tip
DMSO	Dimethyl Sulfoxide
DWP	Deep Well Plate
EN	European Norm
FCA	Flexible Channel Arm
FES	Finger Exchange System
FSE	Field Service Engineer
LDT	Liquid Displacement Technology
LLD	Liquid Level Detection
MCA	Multiple Channel Arm
MCH	Multiple Channel Head
MP	Microplate
MS	MultiSense
PC	Personal Computer
PMP	Pressure Monitored Pipetting
RGA	Robotic Gripper Arm / standard
RGA-Z	Robotic Gripper Arm / long Z
RWP	Rapid Wash Pump
SVN	Subversion (Open source from Apache)
TES	Disposable Tip Ejection System

Abbreviation	Definition
USB	Universal Serial Bus

Glossary

Accuracy

The degree of closeness of measurements to a standard or true value. It is expressed as a percentage, represented by the function: Difference between expected value and actual value, divided by the expected value, multiplied by 100%.

Application

Generally refers to a specific procedure or test, such as RIA (Radio Immuno Assay), EIA (Enzyme Immuno Assay), etc. A corresponding method is programmed in the relevant application software and the application is performed with defined samples and reagents.

Base Worktable

All carriers, labware and tools that are not saved only with the script. In general all carriers and tools are by default a part of the base worktable. The assignment to the base worktable can be altered with the shortcut menu of any active worktable object. Base worktables can be shared between scripts.

Batch

A batch is one part of a method and consists of scripts.

Breakpoint

The system will pause before the execution of the command with a breakpoint. The Operator must press continue on the touch interface or run-time controller.

Capacitive Liquid Level Detection (cLLD)

This is a standard feature of the liquid handling channels of the flexible channel arm. It measures the capacitance between the pipetting tip and the electrical ground (instrument

deck). As soon as the tip touches the liquid, the change in capacitance triggers a signal, which is used for liquid detection. The special features of the cLLD also monitor the tip for an exit signal during aspiration and when the tip retracts from the liquid. The function only works with conductive tips (and DiTis) and conductive liquids.

Carrier

The generic term for components that can be placed on the instrument's deck. Carriers are categorized as nest segments, deck segments, runners (placed on deck segments) and special deck components (e.g. disposable tip waste station, combo carriers). A carrier is a removable support on the instrument's deck. You can use a carrier to hold one or more pieces of labware in place. Carriers can be fitted to the deck at pre-defined grid positions (positioning pins/clips). The pre-defined grid positions are spaced 25.0 mm apart in X-direction. By default, carriers are saved in the base workspace, labware is saved in scripts. NOTE: Carriers are also used to represent positions on devices which can be accessed by a robot. For example, a robot can put a micro plate onto the shuttle of the Te-Stack stacker when the shuttle is at the transfer position. The carrier name indicates the labware type for which it is designed (for device carriers, it indicates the device type).

Carry Over

Residue of any liquid that remains in a tip after rinsing at the end of a pipetting cycle. Such residue is "carried over" to the next cycle. Where carry over needs to be kept to a minimum, disposable tips (DiTis) must be used.

Cleaner

The position in the wash station into which a tip is placed during a wash cycle. System liquid is pumped through the tip. The liquid rising in the cleaner rinses the outer surface of the tips. Overflowing liquid is collected in the waste.

Coefficient of Variation (CV%)

The statistical representation of the precision of a measurement. It is expressed as a percentage, represented by the function: Standard deviation divided by the mean value, multiplied by 100%.

Container

Any vessel placed on or under the deck and containing a liquid or other chemical, e.g. one well in a microplate, a sample tube or a system liquid bottle.

Control

A liquid containing a known concentration of the substance which is to be tested. Used to determine (high/low/cutoff) limits and/or as reference for quality control. The properties of the control are well known and stable.

Deck

Part of the instrument where carriers and options are placed for access by the robotic arm(s).

Diluter

A precision pump used for aspiration and dispense of exactly defined liquid volumes. It mainly consists of a syringe with a motor-driven plunger and a valve to control flow direction. Each pipetting channel is equipped with a diluter.

Disposable tip (DiTi)

Tip often made of plastic that will be discarded after a single aspiration/dispense cycle. Used when carry over needs to be kept to a minimum.

DiTi Transfer Rack

A special rack for the multiple channel arm, which is used to pick up or park the disposable tips.

EMC

Electromagnetic Compatibility

Excess Volume

The volume of excess liquid which is aspirated together (not separated by an air gap) with the liquid to be distributed. It is not dispensed anywhere, but discarded to waste (or a special position) after use, and serves to minimize dilution of the reagents by the system liquid.

Field Service Engineer (FSE)

A person especially trained by the manufacturer for the performance of maintenance and service jobs on the instrument. Also carries out the initial installation of the instrument and its options as well as product upgrades at the user's site.

Fixed Tip

General term for a washable and reusable tip for aspiration/dispense that is fixed to the pipetting device.

Flexible Channel Arm (FCA)

Robotic arm positioned above the instrument's deck area. It holds the pipetting tips and controls the liquid handling. Robot arm on the pipetting instrument which is mounted on the X-slide and typically holds four or eight pipetting tips. It is used for pipetting samples in different volume ranges, depending on the tip types used and the features of the liquid system.

Free Dispense

A dispense mode in which liquid is dispensed while the pipetting tip does not touch the liquid surface in the container. In this mode carry over is minimal, however, droplets may remain sticking to the tip due to surface tension of the liquid.

FSE

Field Service Engineer

GLP

Good Laboratory Practice

Labware

Disposable item such as a microplate or a tube, able to hold liquid in an arrangement of containers or cavities; placed on a carrier for pipetting. Labware can be moved using one of the robotic arms on the instrument.

LIMS

Laboratory Information Management System

Liquid Class

Script defining a theoretical model for a particular type of liquid; includes all liquid handling parameters which are required to pipette liquids of that type. A set of properties defining a theoretical model of a liquid for a specific pipetting device. The liquid classes are set up in the application software. Standard liquid classes are identified by a generic name and include all default liquid handling parameters required to properly handle these liquids. The application specialist may define new liquid classes for optimization or liquids with special properties.

Location

A logical group of sites that can split a carrier or combine sites.

Method

A method is a collection (sequence) of batches. The method can be executed in a run.

Microplate

A labware in the shape of a rectangular plate of standardized size and layout, mostly comprising 96, 384 or 1536 cavities (wells).

Multi Pipetting

The pipetting mode in which a larger amount of liquid is aspirated and then several aliquots are dispensed to different positions.

Nest Segment

A nest segment is a mount for microplates or other labware. It is placed on the instrument's deck.

Pipetting Cycle

A sequence of aspiration and dispense steps to complete a pipetting task.

Pipetting Mode

Describes the main method by which a liquid can be distributed: either by single pipetting or multi pipetting.

Position

A 3-dimensional numbering of positions within a location.

RGA

Robotic Gripper Arm

Robotic Gripper Arm (RGA)

Robotic arm which can be equipped with different types of gripper fingers to pick up, transport and place objects (plates and tubes) within the working area of the pipetting instrument.

Run

A run is the execution of a method. Only one run can be executed at a time.

Runner

Holds sample tubes or reagent troughs. The runners are placed on a deck segment and can slide along the positioning pins, e.g. for barcode identification.

Script

A script is a collection of commands to carry out the actions and interactions required for an assay protocol. More technically speaking it is a sequence of statements and can be executed in a run. Control structures can influence the order of execution of the statements.

Segment

A plate with positioning pins. It is placed on the instrument's deck and positions the runners.

Site

A continuous numbered 1-dimensional identification of a position on a carrier.

Well

One of the cavities in a microplate.

Z-dispense

The height of the point of the pipetting tip at which liquid is dispensed.

Z-max

The lowest possible position the pipetting tip is allowed to reach. During a "search liquid command" the instrument will search for liquid from Z-start down to Z-max. If the tip reaches Z-max without finding liquid, the instrument reacts according to the liquid detection error mode selected.

Z-start

The height of the pipetting tip at which the cLLD is switched on during a "search liquid command". It is usually slightly above the rim of the liquid container.

Z-travel

The height at which the pipetting tip moves from one X/Y-position to another. Moves passing over different labware always use the highest Z-travel defined.