VWorks4 Software

User Guide and Addendum

Agilent Technologies

Notices

User Guide Part Number

G5415-90000

January 2009 rebranded edition
August 2007 addendum edition
August 2006 user guide edition

Contact Information

Agilent Technologies Inc.
Automation Solutions
5301 Stevens Creek Blvd.
Santa Clara, CA 95051
USA

Technical Support: 1.800.979.4811
or +1.408.345.8011
service.automation@agilent.com

Customer Service: 1.866.428.9811
or +1.408.345.8356
orders.automation@agilent.com

European Service: +44 (0)1763853638
euroservice.automation@agilent.com

Documentation feedback:
documentation.automation@agilent.com

Web:
www.agilent.com/lifesciences/automation

Acknowledgements

Microsoft and Windows are registered trademarks of the Microsoft Corporation in the United States and other countries.

Warranty

The material contained in this document is provided "as is," and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as "Commercial computer software" as defined in DFAR 252.227-7014 (June 1995), or as a "commercial item" as defined in FAR 2.101(a) or as "Restricted computer software" as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Agilent Technologies' standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Safety Noticies

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

A **CAUTION** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a **CAUTION** notice until the indicated conditions are fully understood and met.

Letter to our Customers

Dear Customer,

The Agilent Technologies acquisition of Velocity11 resulted in the following changes:

- Creation of Agilent Technologies Automation Solutions, formerly Velocity11
- Renaming of some Velocity11 products
- New Customer Service and Technical Support contact information
- New website address for product information

Please make a note of the following changes as they impact this user guide.

Velocity11 product name changes

Velocity11 product name	Changes to ...
Access2 Automated Microplate Loader	Automated Centrifuge Loader
Element Automation System	BioCel 900 System
IWorks Device Driver Programming Interface	VWorks Device Driver Interface
PlatePierce Seal Piercing Station	Microplate Seal Piercer
VCode Barcode Print and Apply Station	Microplate Barcode Labeler
Velocity11 Robot	3-Axis Robot
VHooks Integration Interface	VWorks Hooks Interface
VPrep Pipetting System	Vertical Pipetting Station
VSpin Microplate Centrifuge	Microplate Centrifuge
VStack Labware Stacker	Labware Stacker

New contact information

Documentation feedback: documentation.automation@agilent.com

Technical Support: 1.800.979.4811 or +1.408.345.8011
service.automation@agilent.com

Customer Service: 1.866.428.9811 or +1.408.345.8356
orders.automation@agilent.com

European Service: +44 (0)1763853638
euroservice.automation@agilent.com

Web: www.agilent.com/lifesciences/automation

Contents

About this addendum 1

Installing the VWorks4 software 3

Creating a protocol. 7

Using the Location, location property. 23

Specifying pipette location offsets. 27

Using an m x n array of pipette tips 37

Tracking pipette tip usage 47

Automating labware stacking on the Bravo deck. 57

Prompting users for task parameter values 67

Using JavaScript utilities. 79

Using the VWorks4 ActiveX control. 85

About this addendum

This addendum contains the following topics:

- *Installing the VWorks4 software*. Explains how to install the VWorks4 software.
- *Creating a protocol*. Describes the workflow for creating a protocol.
- *Using the Location, location property*. Describes the available settings for the Location, location property and explains how to automate plate placement.
- *Specifying pipette location offsets*. Describes the Pipette Technique property for Bravo pipetting tasks.
- *Using an m x n array of pipette tips*. Describes how to specify the use of an m × n array of pipette tips.
- *Tracking pipette tip usage*. Describes how to track pipette tip usage during and across protocol runs.
- *Automating labware stacking on the Bravo deck*. Describes how to stack labware on the Bravo deck for multiple-labware processing.
- *Prompting users for task parameter values*. Describes how to add variables to prompt users for parameter values at the beginning or at desired points during the protocol run.
- *Using JavaScript utilities*. Describes two JavaScript utilities: ActiveX Wrapper and File Object.
- *Using the VWorks4 ActiveX control*. Describes the use of the VWorks4 ActiveX control when integrating third-party software.

Installing the VWorks4 software

This section contains the following topics:

❑ "Uninstalling existing VWorks4 software" on page 4
❑ "Installing the VWorks4 software" on page 5

Uninstalling existing VWorks4 software

About this topic

If you have an existing version of the VWorks4 software, you must first uninstall it before installing the new software. This topic explains how to uninstall the existing software.

Procedure

To uninstall an existing version of the VWorks4 software:

1. On the **Start** menu, select **Settings > Control Panel.** The Control Panel window appears.
2. Double-click **Add/Remove Programs.** The Add or Remove Programs dialog box opens.
3. Locate and select VWorks or VWorks_buildx.x.x, and then click **Change/Remove.** The removal process might take a couple of minutes.

   When the removal process is finished, the following folders and files remain in the C:\Program Files\Velocity11\VWorks folder or where you previously installed the software:

   - Logs folder
   - Users folder
   - VWorks.vln file

Installing the VWorks4 software

About this topic

This topic explains how to install the VWorks4 software. If you have an existing version of the software on your computer, you must uninstall it first. For instructions, see “Uninstalling existing VWorks4 software” on page 4.

Procedure

To install the new VWorks4 software:

1. Insert the VWorks4 software CD into the computer's CD-ROM drive. The software installer should start automatically. If it does not, navigate to the CD-ROM drive, and then double-click VWorks_8.0.0_TBD.exe.
2. Follow the instructions on the installation software screens.
3. In the last screen, select **Launch VWorks Software Activation Program** if this is a new VWorks installation (you did not have an existing version on the computer).
4. Click **Finish.** If you selected Launch VWorks Software Activation Program, the VWorks Software Activation dialog box opens. Follow the instructions in the dialog box to obtain a software license file.
5. Place the license file (VWorks.vln) in the folder that contains the VWorks4 software. For example, if you installed the software in the C:\Program Files\Velocity11\VWorks folder, place the VWorks.vln file in that folder.

Creating a protocol

This section contains the following topics:

- ❑ "Preparing for protocol writing" on page 8
- ❑ "Workflow for creating a protocol" on page 11
- ❑ "Configuring labware" on page 12
- ❑ "Setting the plate properties for configured labware" on page 15
- ❑ "Setting task parameters" on page 17
- ❑ "Converting configured labware to a protocol process" on page 21

Preparing for protocol writing

About this topic

Before you create a protocol, you need to determine:

- The accessories you need for the protocol and where to place them on the Bravo deck.
- The labware that will be used or processed during the protocol run and their starting and ending deck locations.

This topic explains how you specify your plan for the different accessories and labware in a protocol and the terminology that is used.

Planning accessory use

After you determine the accessories you need for the protocol run, you should specify their locations on the Bravo deck. You do this when you configure the accessory in the Bravo Diagnostics Configuration tab. For instructions, see the *Bravo User Guide*.

!! IMPORTANT !! Accessory locations are displayed in the Bravo Diagnostics Configuration tab only. You need to remember their locations when configuring labware in the protocol.

Note: Accessories stay at the same deck location throughout a protocol run.

For the list of available accessories, see the *Bravo User Guide* or the Velocity11 website at www.velocity11.com.

Planning labware use

In a protocol run, labware can either:

- Transfer onto and off the Bravo deck for processing.
- Start and stay on the Bravo deck during the entire protocol run.

When you plan a protocol, you should determine how the labware will be used or processed and how they will move on the Bravo deck. For example, if you are writing a microplate replication protocol, you need to decide whether the source microplates or destination microplates will be moved onto the deck while the other will remain stationary on the deck. The decision can depend on many factors, including your preferences.

In the VWorks4 software, labware can be categorized as follows:

- Process plates
- Configured labware
- Static labware

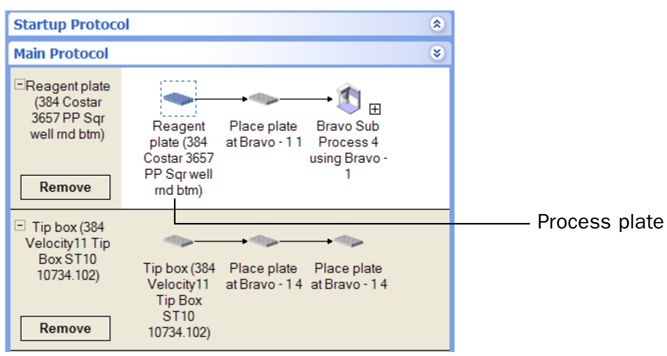
Process plates

A process plate is a labware that:

- Is transferred onto the Bravo deck automatically or manually during the protocol run.
- Is the object of one or more tasks in the protocol run.
- Might move to different deck locations during the run.
- Is transferred off of the deck automatically or manually during the protocol run.

In the software, one process plate can represent a number of physical microplates or labware that will be processed during a protocol run. For example, in a microplate replication protocol, the process plate can represent 10 destination microplates that are transferred sequentially onto the Bravo deck.

A process plate is displayed as the first icon in a protocol process, a process in which you can add or remove tasks. The protocol process has a white background.



For more information about process plates, see the *VWorks4 User Guide*. To create a protocol process, see “Workflow for creating a protocol” on page 11.

Configured labware

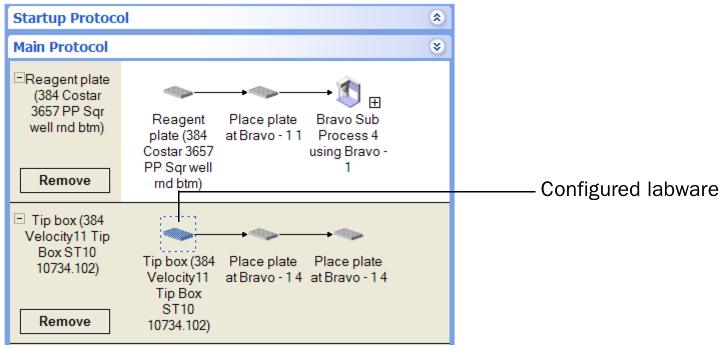
A configured labware is a labware that:

- Is on the Bravo deck before the protocol run starts.
- Is used by one or more tasks in the protocol process.
- Might move to different deck locations during the run.
- Remains on the deck after the protocol run is finished.

In the software, configured labware represents the single physical labware on the deck, such as a tip box. For example, if you are using two different tip boxes in a protocol, you would configure two labware in the software, one for each tip box on the deck.

Like accessories, you must let the software know the labware’s starting location on the Bravo deck. To do this, you need to configure the labware. For instructions, see “Configuring labware” on page 12.

Configured labware is displayed in a protocol with a gray background. If it is used by a task in a Bravo sub-process, a copy of the sub-process tasks are shown next to the configured labware. You cannot add or remove tasks in the duplicated process. However, whenever the sub-process is updated, the duplicate copy is also updated automatically.



You have the option of converting a configured labware into a process plate. For instructions, see “Converting configured labware to a protocol process” on page 21.

Static labware

!! IMPORTANT !! The latest version of the VWorks4 software is backward-compatible with protocols created in VWorks4 version 6.2.3 or earlier and will continue to support static labware configuration procedures. However, Velocity11 recommends that you use the configured labware when writing new protocols.

A static labware is a labware that will start on the Bravo deck and will remain at the same location during the protocol run. For example, a tip box can be a static labware.

To specify its starting location, you must configure the static labware using one of the following methods:

❑ The Bravo Sub-Process task in a Main Protocol

❑ The Configure Static Labware task in the Startup Protocol

In general, you configure static labware before the first task in a protocol. If you have multiple processes in the protocol, configure the labware once before the first task of the first process.

Configure labware in a Startup Protocol if the labware configuration will be used in all the Main Protocol subprocesses. Configure labware in the Main Protocol if you want to override the labware configuration in the Startup Protocol.

Workflow for creating a protocol

About this topic

This topic outlines the revised procedures you need to follow to create a protocol.

Workflow

The following table presents the steps for creating a protocol.

Step	For this task...	See...
1	Create a new protocol.	*VWorks4 User Guide*
2	Set protocol options.	*VWorks4 User Guide*
3	Configure labware.	"Configuring labware" on page 12
4	Set plate properties for the configured labware.	"Setting the plate properties for configured labware" on page 15
5	Set plate properties for the protocol process.	*VWorks4 User Guide*
6	Add tasks to the protocol process and set the task parameters.	*VWorks4 User Guide* and "Setting task parameters" on page 17
7	Set up startup and clean up protocol processes.	*VWorks4 User Guide*
8	Save the protocol.	*VWorks4 User Guide*
9	Compile the protocol.	*VWorks4 User Guide*
10	Simulate a protocol run.	*VWorks4 User Guide*

Configuring labware

About this topic

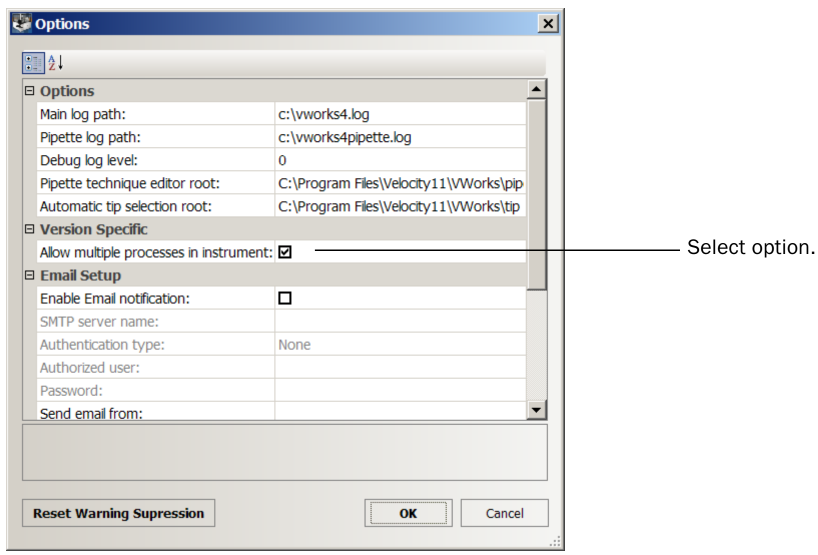
After you create a new protocol, you can configure labware that will remain on the Bravo deck before and after processing. This topic explains how to configure labware.

For a review of the protocol workflow, see “Workflow for creating a protocol” on page 11.

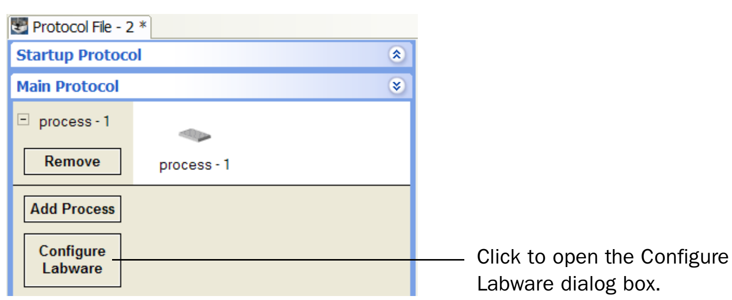
Procedure

To configure labware:

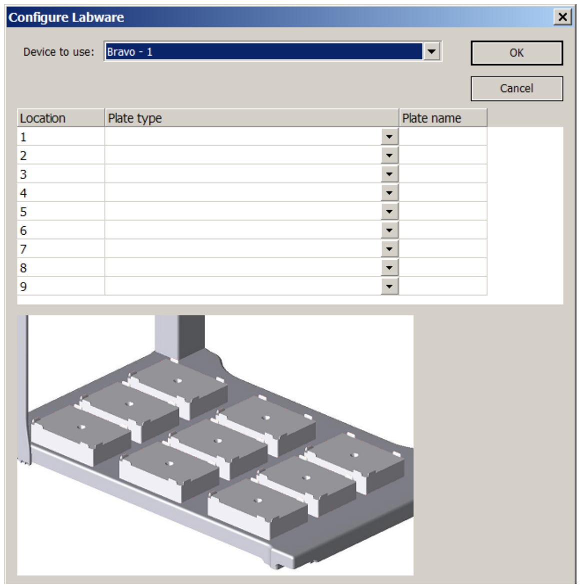
1. In the VWorks4 Instrument software, make sure the multi-process protocol option is turned on. To do this:
   a. On the **Tools** menu, click **Options.** The Options dialog box opens.
   b. Select **Allow multiple processes in instrument.**



2. In the **Main Protocol,** in the area below the new protocol process, click **Configure Labware.**

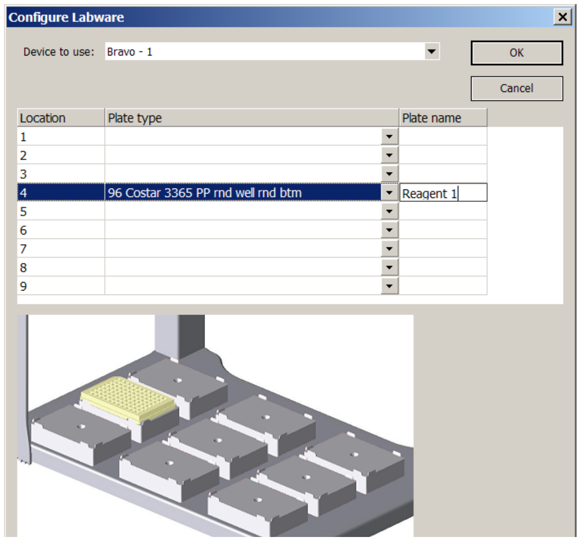


The Configure Labware dialog box opens.



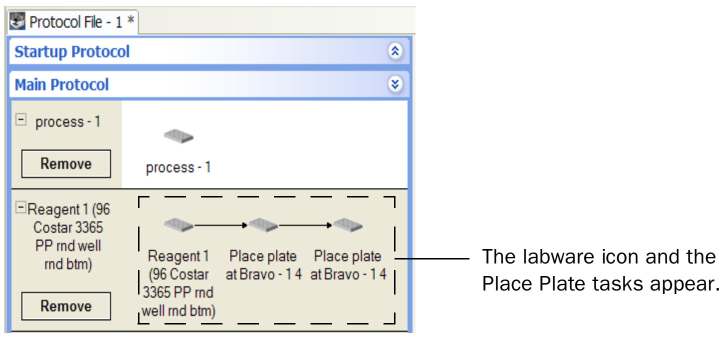
3. In the **Device to use** list, select the Bravo device on which you want to run the protocol.
4. In the **Location** table:
   a. At the deck location number, double-click the **Plate name** box, and then type a name for the labware at the deck location.
   b. Select the labware for the corresponding deck location. To do this, click the box next to the plate name to display the list, and then select the desired labware. The labware appears in the graphic below the table.

      Note: Configured static labware (from protocols created in VWorks4 version 6.2.3 or earlier) will appear in the table and graphic. However, configured accessories do not appear in the table and graphic.

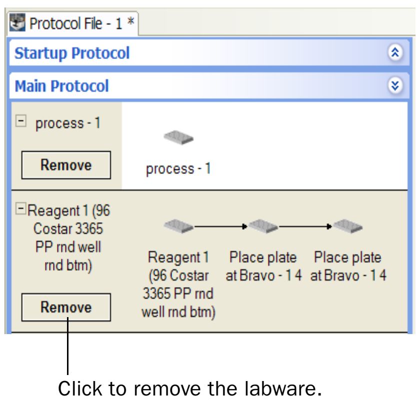


5. When you are finished, click **OK** to save the information and close the Labware Configuration dialog box.

   In the Main Protocol area, the deck labware appears. Click + to expand and view the labware icon. Notice that two Place Plate tasks are automatically added, one to indicate its starting location and the other to make sure it returns to its starting location at the end of the protocol.



6. If you want to remove the deck labware, click **Remove.**



Setting the plate properties for configured labware

About this topic

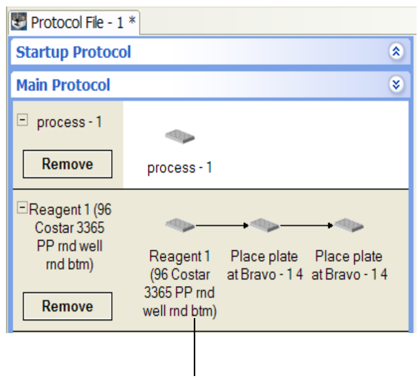
After you configure a labware, you need to set its plate properties. This topic explains how to set the plate properties for configured labware.

For a review of the protocol workflow, see “Workflow for creating a protocol” on page 11.

Procedure

To set the plate properties for a deck labware:

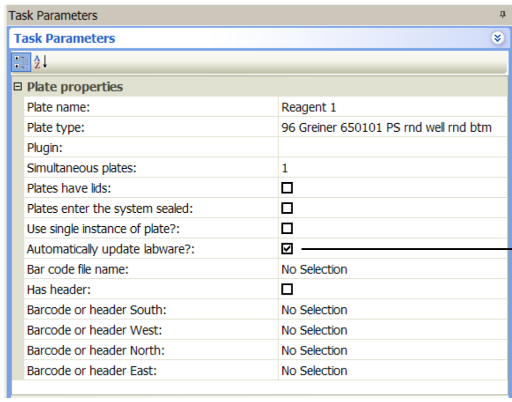
1. In the **Main Protocol** area, select the labware icon.



Select the labware icon.

2. In the **Task Parameters** area, set the **Plate properties.** The properties are identical to the plate properties for a protocol process. See the *VWorks4 User Guide* for the list and description of the protocol process plate properties.
3. In the **Task Parameters** area, select **Automatically update labware** if the labware will be used by one or more tasks in the protocol process. Selecting the option ensures that when you revise the protocol, the changes are automatically reflected in the configured labware’s process.

Clearing the check box converts the labware to a protocol process. For more information, see “Converting configured labware to a protocol process” on page 21.



Select to ensure changes in the protocol process are reflected in the labware’s process.

Setting task parameters

About this topic

After you configure labware and set its plate properties, you can add tasks to the protocol process. See the *VWorks4 User Guide* for instructions.

When you set the parameters for a task, you can select the configured labware you want to use for the task. If the labware is used in two tasks that are in two different protocol processes, you can specify the sequence.

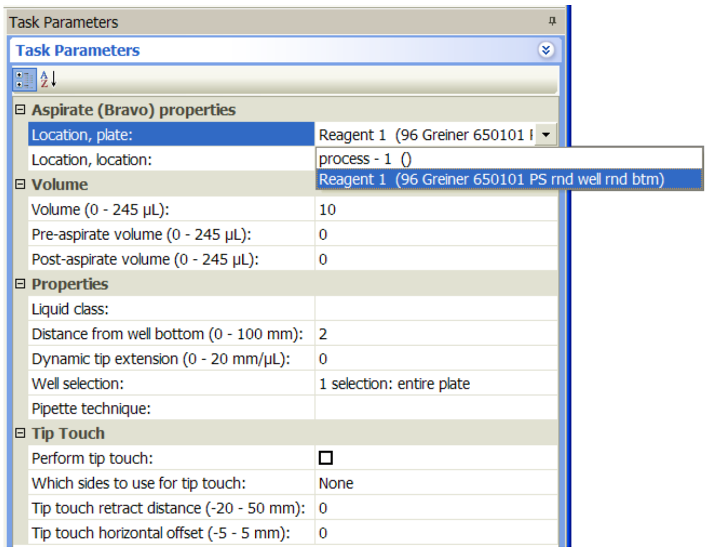
This topic explains how to specify a configured labware in a task.

For a review of the protocol workflow, see "Workflow for creating a protocol" on page 11.

Specifying a configured labware in a task

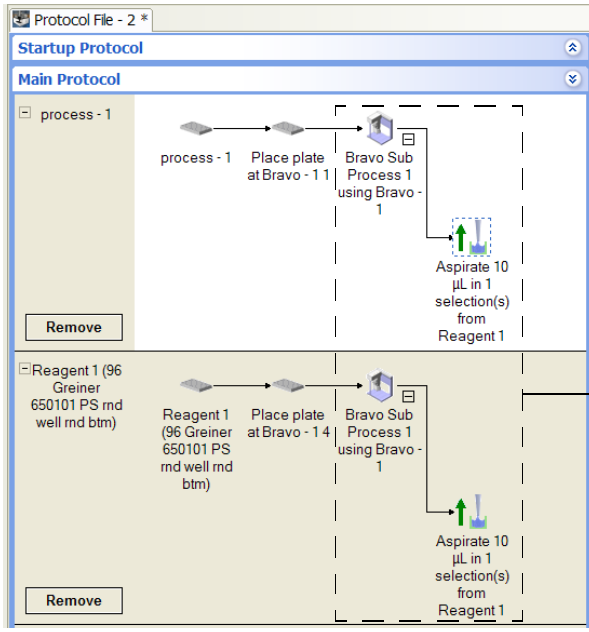
To specify a configured labware in a task:

1. In the protocol process, select a task.
2. In the **Task Parameters** area, select the configured labware you want to use in the **Location, plate** list.



In the Main Protocol area, the subprocess containing the task is copied into the configured labware process area.

Note: The link is only possible if you selected the Automatically update labware option in the labware's plate Properties. For information, see "Setting the plate properties for configured labware" on page 15.



The duplicate copy of the subprocess indicates the configured labware is used in a task.

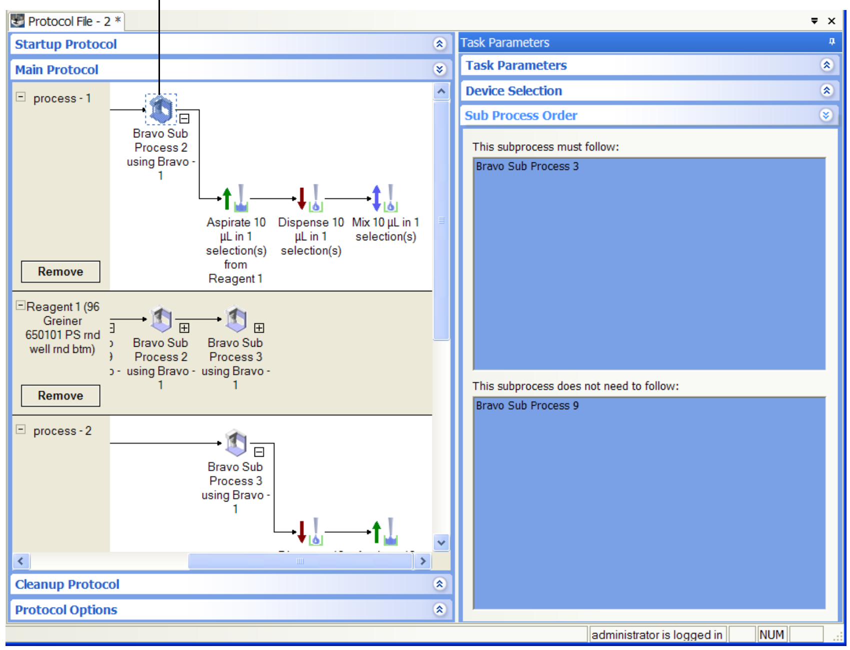
Specifying the task sequence

If more than one sub-process uses the same configured labware, and the sub-processes are in different protocol processes, you can specify the sequence in which the sub-processes will be performed.

To specify the sequence in which the sub-processes will be performed on the same configured labware:

1. In the protocol process, select the Bravo sub-process that contains the task that uses the configured labware.

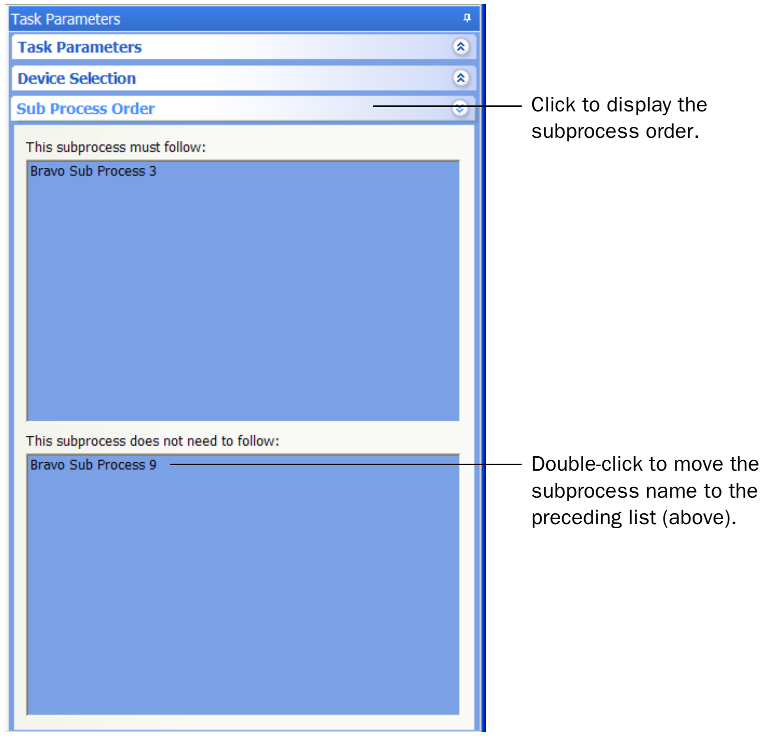
Select the desired Bravo sub-process.



2. In the **Task Parameters** area, click **Sub Process Order.**

3. In the **Sub Process Order** area, double-click the subprocess names to rearrange the order.

   In the example shown, the task sequence will be SubProcess 9, current subprocess, and SubProcess 3.



Converting configured labware to a protocol process

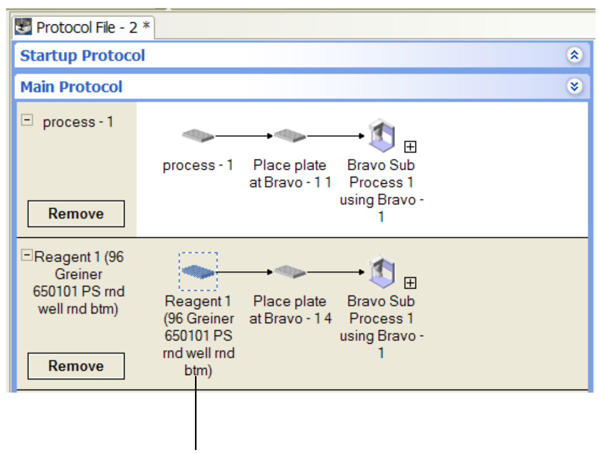
About this topic

This topic explains how to convert a configured labware to a protocol process.

Procedure

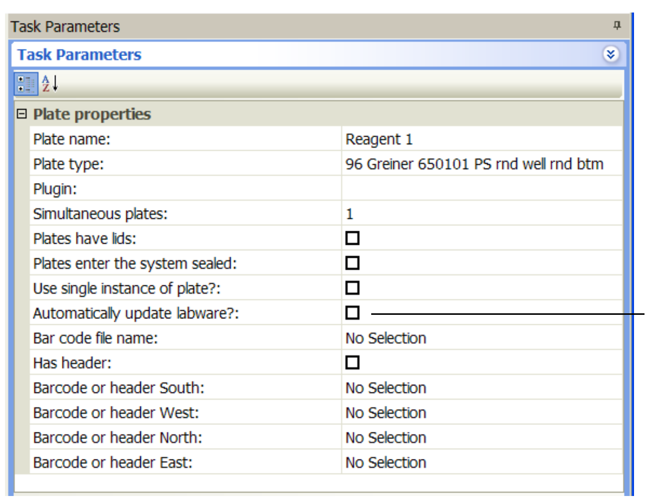
To convert a configured labware to a protocol process:

1. In the **Main Protocol** area, select the configured labware icon.



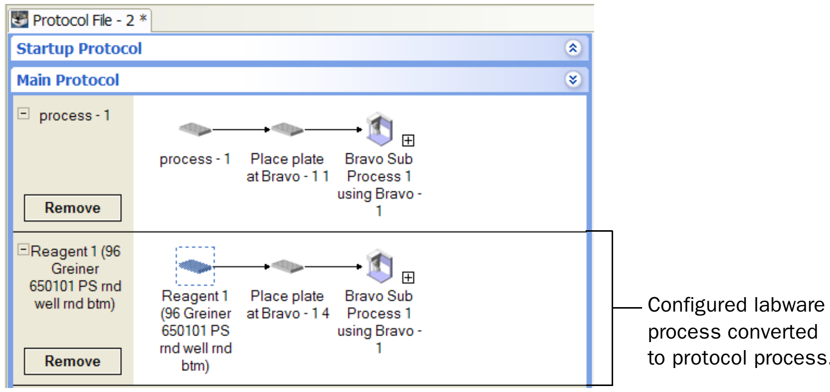
Select the configured labware you want to convert.

2. In the **Plate Properties** area, clear the **Automatically update labware** check box.



Clear the check box.

The background in the configured labware process area becomes white, indicating that it is converted to a protocol process.



3. Edit the newly converted protocol process.

Using the Location, location property

This section contains the following topics:

❑ "About the Location, location property" on page 24
❑ "Setting the Location, location property" on page 25

About the Location, location property

About this topic

This topic describes the available Location, location selections in the VWorks4 software.

Description

Whenever you add a Bravo pipetting task to a protocol process, you must set the Location, location property to tell the VWorks4 software at which location the task will occur. The following selections are available for the Location, location property:

- <*auto-select*>. Enables the Bravo robot to automatically place the plate at the first-available or appropriate location for the task. If accessories are installed on the Bravo deck, the VWorks4 software uses the accessory configuration information in Bravo Diagnostics to determine the correct deck location for the task.
- *Location number*. Allows you to manually select the location for the task. You manually select a deck location if you do not want the VWorks4 software to automatically select a location. If you have two identical accessories on the Bravo deck, you can manually select a location to direct the robot to one of the accessories.

  Note: If the manually selected location does not have the correct accessory, the software will display an error message when the protocol is compiled.

Setting the Location, location property

About this topic

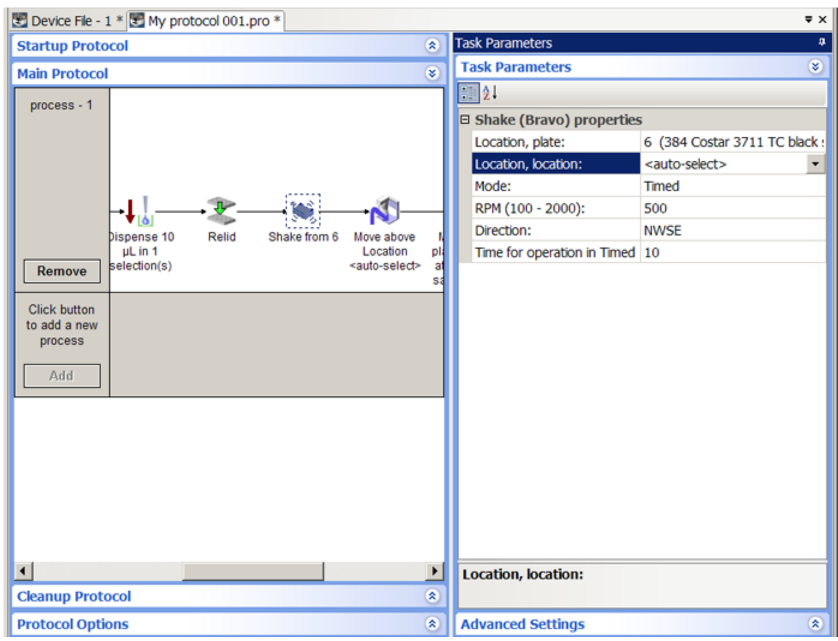
This topic explains how you set the Location, location property.

Procedure

To set the Location, location property:

1. In the VWorks4 window, select a pipetting task in the protocol process.
2. In the **Task Parameters** area, in the **Location, location** list, select one of the following:
   - **<auto-select>**
   - Location number **1, 2, 3, ...,** or **9**

   Note: <auto-select> is the default selection.



Specifying pipette location offsets

This section contains the following sections:

- ❑ "About Pipette Techniques" on page 28
- ❑ "Creating and editing Pipette Techniques" on page 28
- ❑ "Selecting a Pipette Technique in a task" on page 33
- ❑ "Managing existing Pipette Techniques" on page 34
- ❑ "Storing Pipette Technique files" on page 35
- ❑ "Refining or troubleshooting Pipette Techniques" on page 36

About Pipette Techniques

About this topic

This topic explains the use of Pipette Techniques to specify different pipetting methods.

Pipette Techniques

You can define a Pipette Technique to use different pipetting methods. Different applications can benefit from different pipetting methods. For example, in multiplexed plates or cell-based assays, dispensing at an offset from the well center can improve distribution of the fluid.

Using the Pipette Technique property

You can define any number of Pipette Techniques. After you create a technique, it becomes available for the following tasks in any protocol:

- Aspirate
- Dispense
- Mix

Creating and editing Pipette Techniques

About this topic

This topic explains how to create a new and edit an existing Pipette Technique.

Creating a Pipette Technique

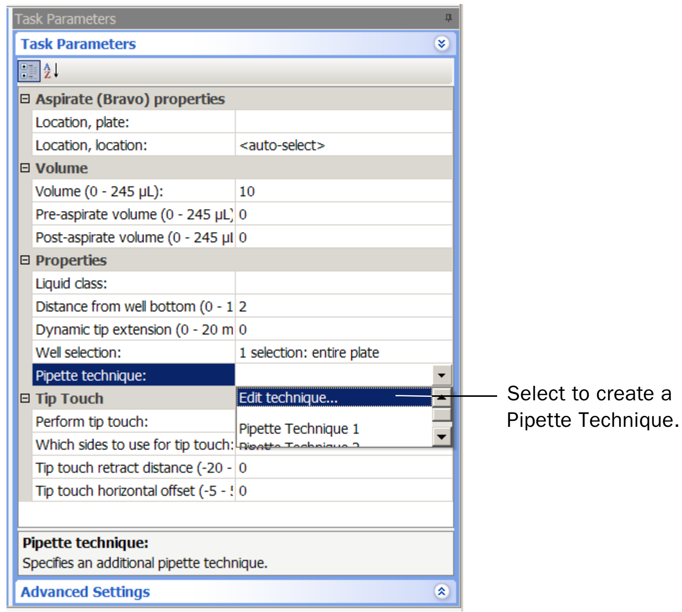
You can create a Pipette Technique in two ways:

- When setting task parameters in a protocol
- Using the Tools menu

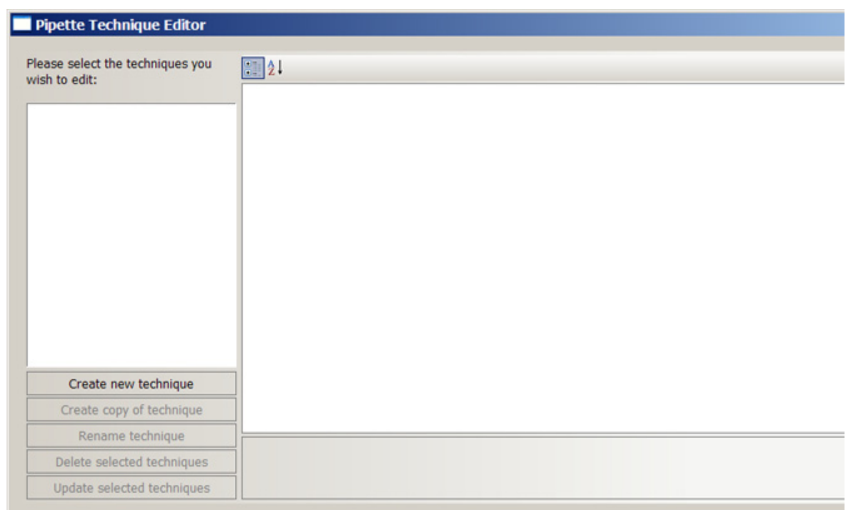
To create a Pipette Technique when setting task parameters:

1. In the VWorks4 window, create a new protocol.
2. Add the SubProcess (Bravo) task.
3. Add one of the following tasks:
   - Aspirate (Bravo)
   - Dispense (Bravo)
   - Mix (Bravo)

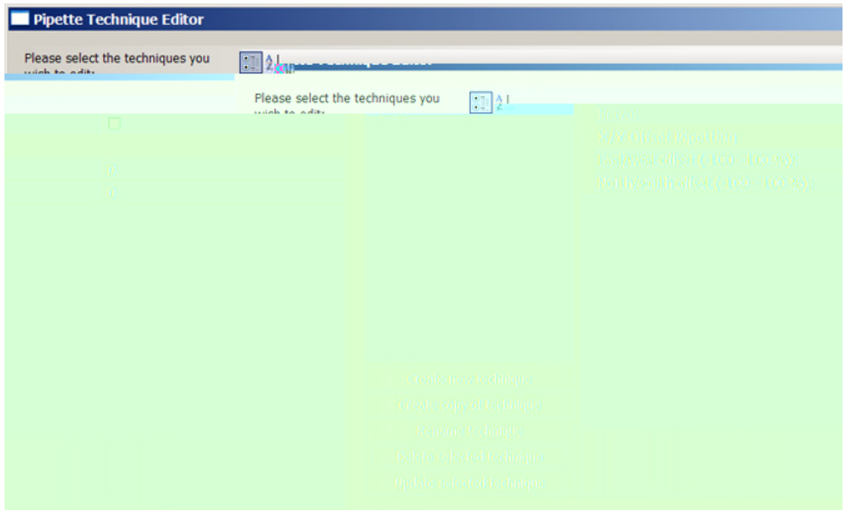
4. In the **Task Parameters** area, select **Edit technique** in the **Pipette Technique** property list.



The Pipette Technique Editor dialog box opens.



5. Click **Create new technique,** type a technique name, and then click **OK.** The new technique name appears in the Pipette Technique Editor dialog box. In addition, the X/Y Offset parameters appear to the right of the technique name.



6. In the **X/Y Offset Pipetting** table, do the following:
   a. In the **East/west offset (-100 – 100%)** box, type the distance (in percent of well radius) you want the pipette to move in the X direction. A 0 value does not move the pipette horizontally. A positive value moves the pipette to the right. A negative value moves the pipette to the left.
   b. In the **North/south offset (-100 – 100%)** box, type the distance (in percent of well radius) you want the pipette to move in the Y direction. A 0 value does not move the pipette forward or backward. A positive value moves the pipette backward (away from the front of the Bravo). A negative value moves the pipette forward (toward the front of the Bravo).
7. When you are finished, click **Update the selected technique.**

To create a Pipette Technique using the Tools menu:

1. In the VWorks4 window, on the **Tools** menu, click **Pipette Technique Editor.** The Pipette Technique Editor dialog box opens.
2. Click **Create new technique,** type a technique name, and then click **OK.** The new technique name appears in the Pipette Technique Editor dialog box. In addition, the X/Y Offset parameters appear to the right of the technique name.
3. Set the parameters in the **X/Y Offset Pipetting** table.
4. When you are finished, click **Update the selected technique.**

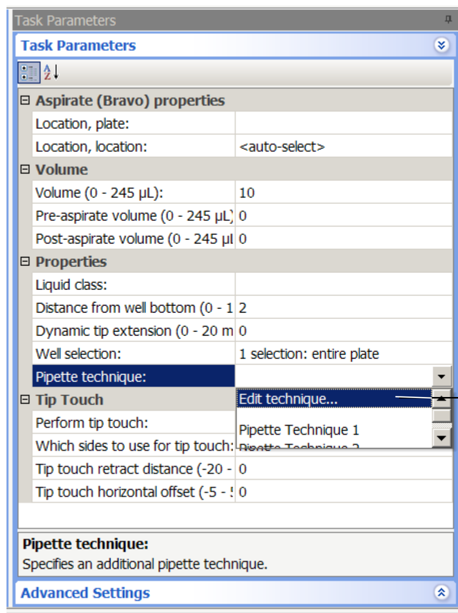
Editing a Pipette Technique

You can edit an existing Pipette Technique in two ways:

- ❑ In a protocol
- ❑ Using the Tools menu

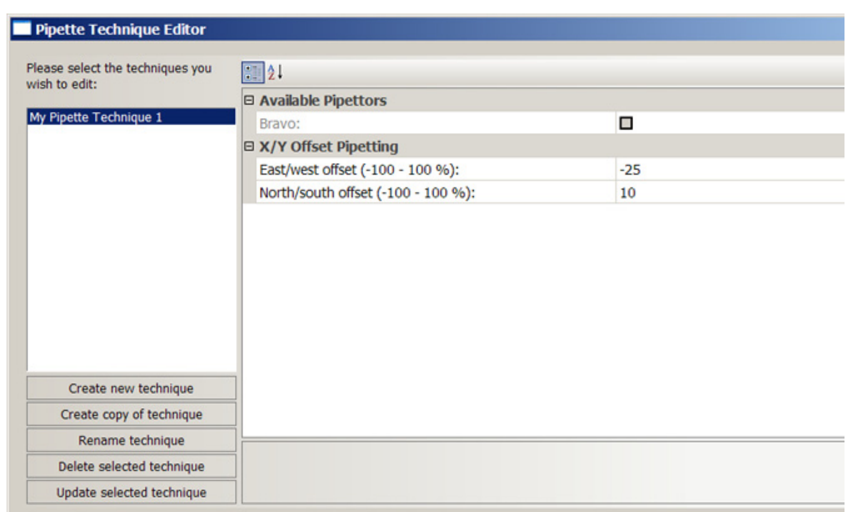
To edit a Pipette Technique in a protocol:

1. In the VWorks4 window, open the protocol.
2. In the Main Protocol area, select the task for which the Pipette Technique will change.
3. In the **Task Parameters** area, select **Edit technique** in the **Pipette Technique** property list.



Select to edit a Pipette Technique.

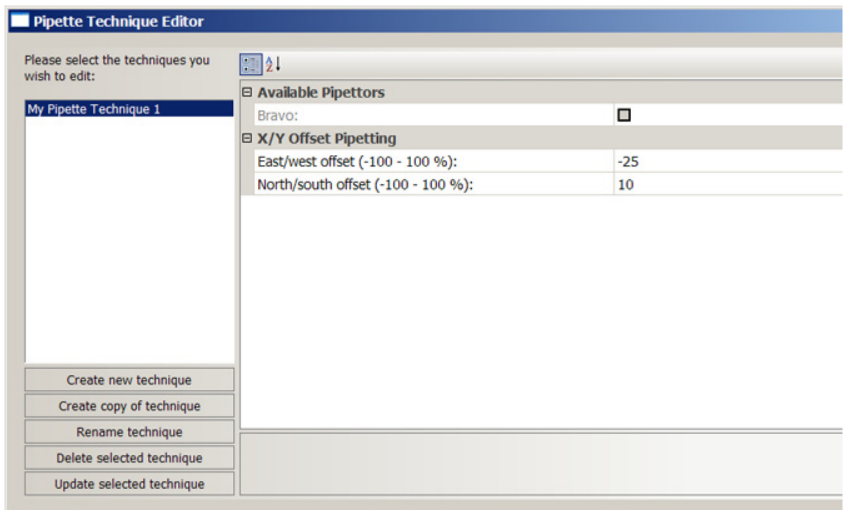
The Pipette Technique Editor dialog box opens.



4. Select the Pipette Technique on the left side of the dialog box.
5. Make the desired changes in the **X/Y Offset Pipetting** table.
6. When you are finished, click **Update the selected technique.**

To edit a Pipette Technique using the Tools menu:

1. In the VWorks4 window, on the **Tools** menu, click **Pipette Technique Editor.** The Pipette Technique Editor dialog box opens.



2. Select the Pipette Technique on the left side of the dialog box.
3. Make the desired changes in the **X/Y Offset Pipetting** table.
4. When you are finished, click **Update the selected technique.**

Selecting a Pipette Technique in a task

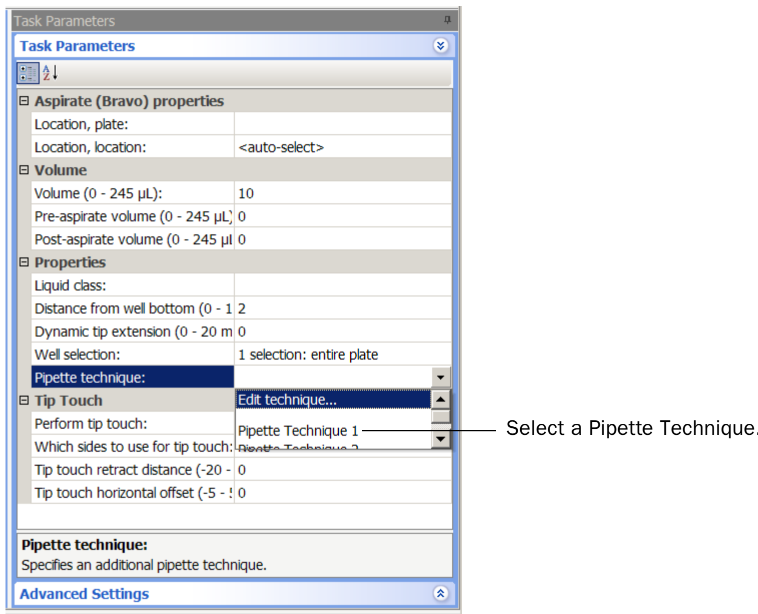
About this topic

This topic explains how to select a Pipette Technique in a pipetting task.

Selecting a Pipette Technique

To select a Pipette Technique in an Aspirate, Dispense, or Mix task:

1. Select the **Aspirate, Dispense,** or **Mix** task in the protocol.
2. In the **Task Parameters** area, select the pipette technique from the **Pipette Technique** property list.



Managing existing Pipette Techniques

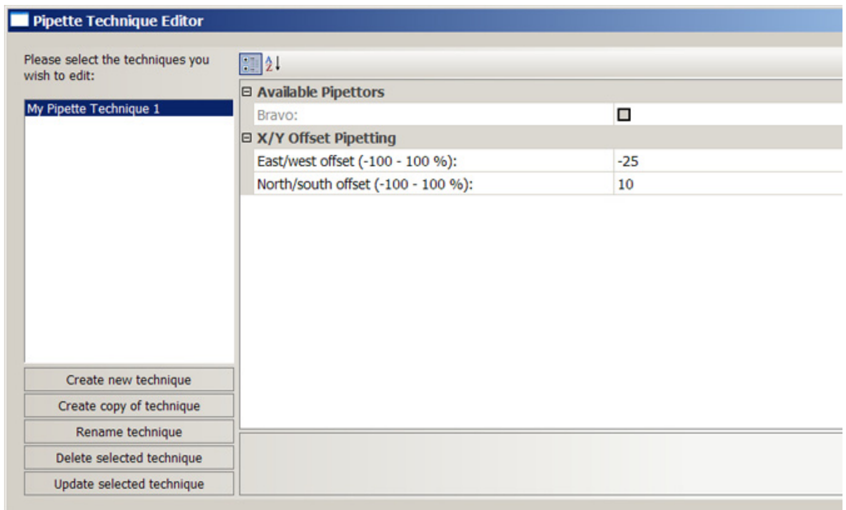
About this topic

This topic explains how to copy, rename, and delete existing Pipette Techniques.

Managing existing techniques

To copy, rename, and delete an existing Pipette Technique:

1. In the VWorks4 window, on the **Tools** menu, click **Pipette Technique Editor.** The Pipette Technique Editor dialog box opens.



2. Select the Pipette Technique on the left side of the dialog box.
3. Click one of the following:
   - **Create copy of technique.** The software prompts you to type a new name for the duplicated technique, and then creates a copy of the selected technique and saves it using the new name.
   - **Rename technique.** The software prompts you to type a new name for the selected technique, and then saves the technique using the new name.
   - **Delete selected technique.** The software deletes the selected technique.

Storing Pipette Technique files

About this topic

This topic explains where the Pipette Technique files are stored and how you can change the storage location.

Default storage location

By default, the VWorks4 software stores Pipette Technique files in the following folder:

...\Velocity11\VWorks\pipette techniques

Changing the storage location

To change the location of the Pipette Technique files:

1. In the VWorks4 software, on the **Tools** menu, click **Options.** The Options dialog box opens.
2. In the **Options** table, click **Pipette Technique Editor root,** and then click the browse button (...). The Browse for Folder dialog box opens.
3. Locate and select a folder for the Pipette Technique files.
4. Click **OK** to save the new location.

Refining or troubleshooting Pipette Techniques

About this topic

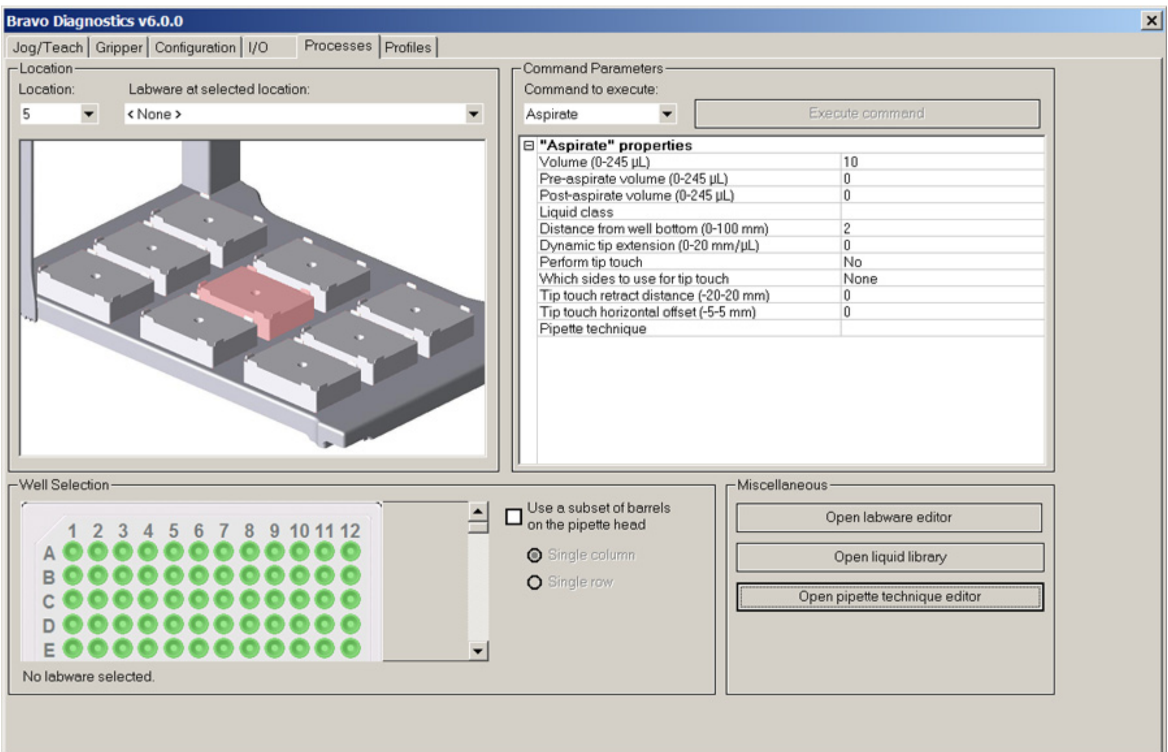
This topic explains how you can access the Pipette Technique Editor from Bravo Diagnostics.

Accessing the Pipette Technique Editor from Bravo Diagnostics

For development and troubleshooting purposes, you can use the Bravo Diagnostics Processes tab to perform the pipetting tasks. In addition to executing pipetting commands from the Processes tab, you can also open the Pipette Technique Editor to refine or troubleshoot pipetting techniques.

To open the Pipette Technique Editor in the Bravo Diagnostics Processes tab:

In the **Miscellaneous** area, click **Open pipette technique editor.**



Using an m x n array of pipette tips

This section contains the following sections:

- "Possible array configurations and limitations" on page 38
- "Setting the Set Head Mode task parameters" on page 42

Possible array configurations and limitations

About this topic

You can install and use an m × n array of pipette tips in protocol runs. The m × n configuration permits complex liquid-handling tasks in a protocol run.

This topic describes the possible array configurations and presents the deck location limitations.

Pipette tip arrays

The VWorks4 software permits the use of an m × n array of tips on the pipette head. The m × n array includes the following configurations:

- All pipette channels
- The first or last full column or row of pipette channels
- Multiple full columns or rows of pipette channels
- A partial number of channels in the first or last column or row
- Multiple partial columns or rows of pipette channels
- A single pipette channel at one of the corners of the pipette head

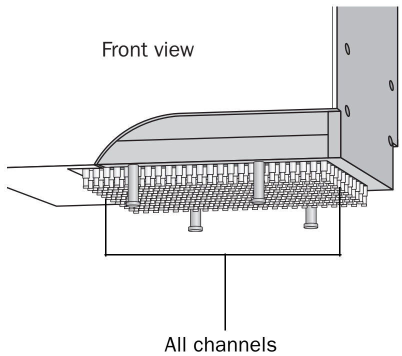
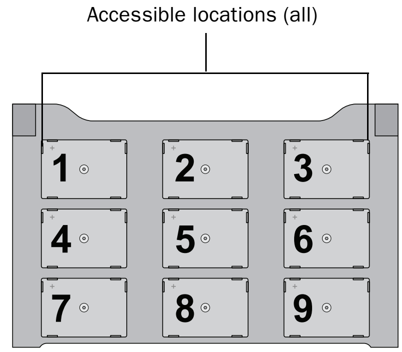
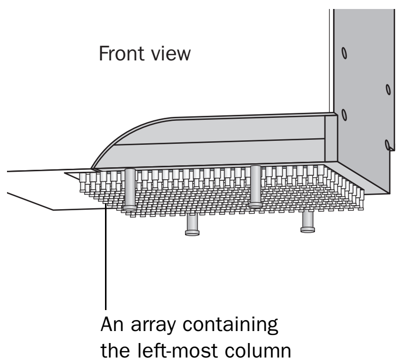
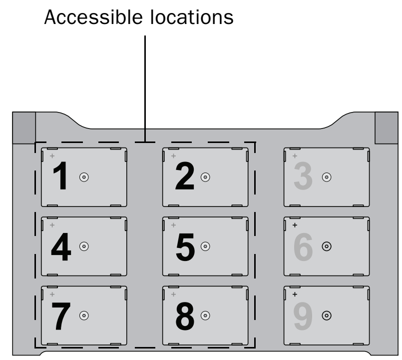
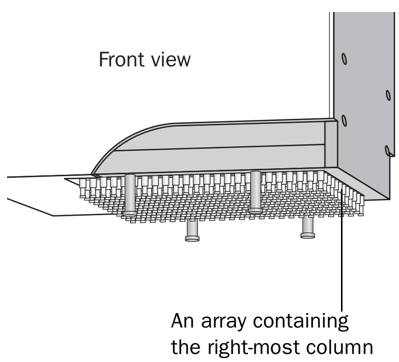
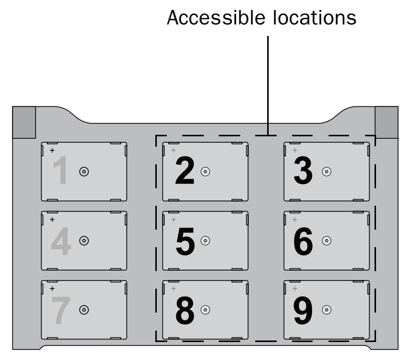
Pipette head requirements

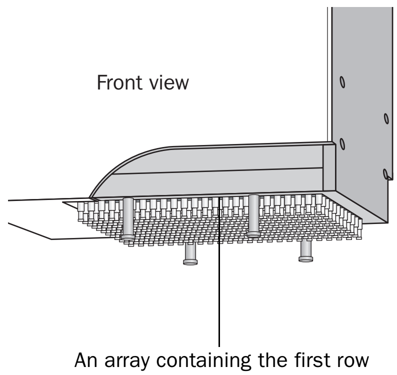
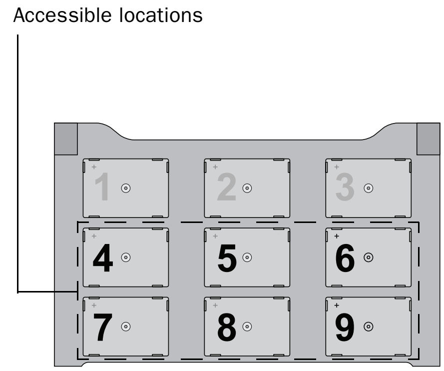
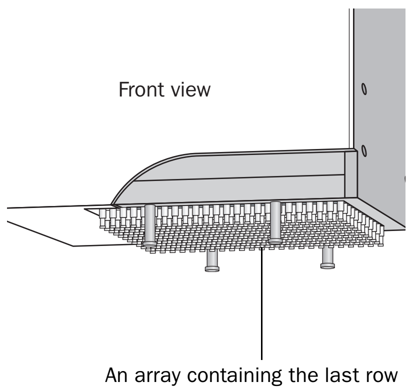
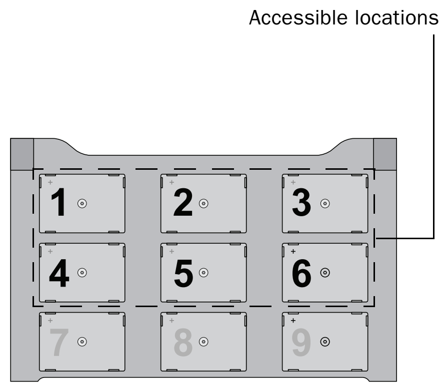
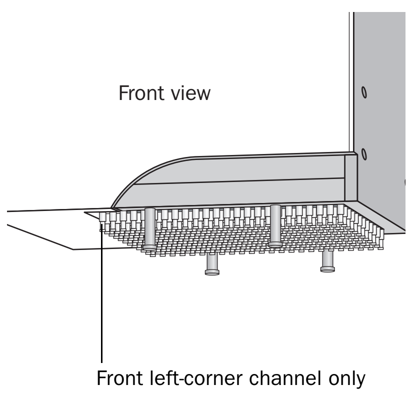
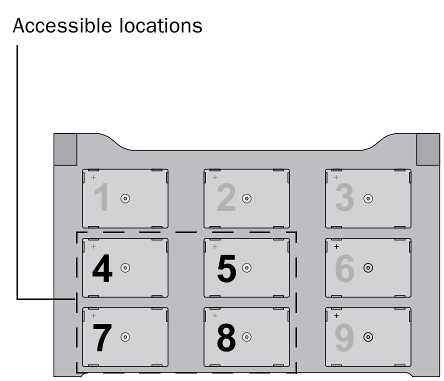
You can use the following pipette heads with a flexible array of pipette tips:

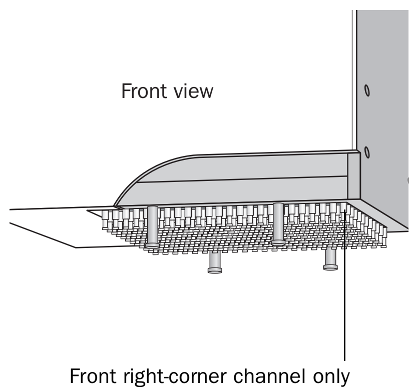
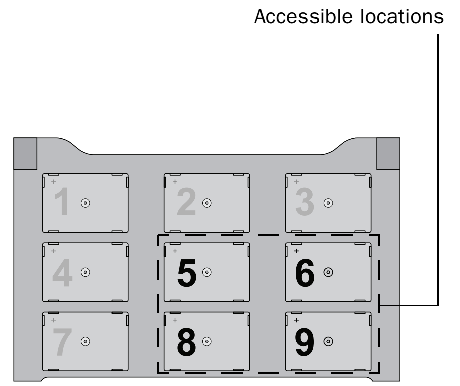
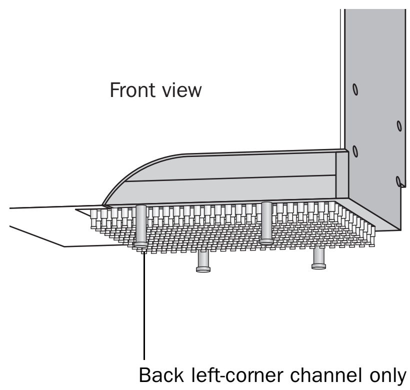
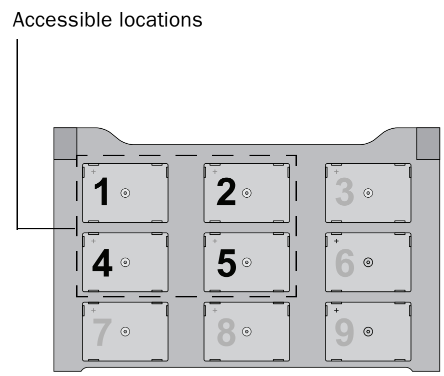
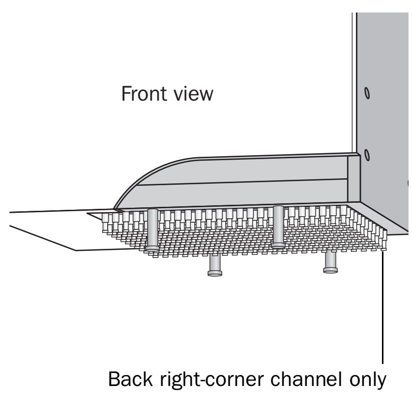
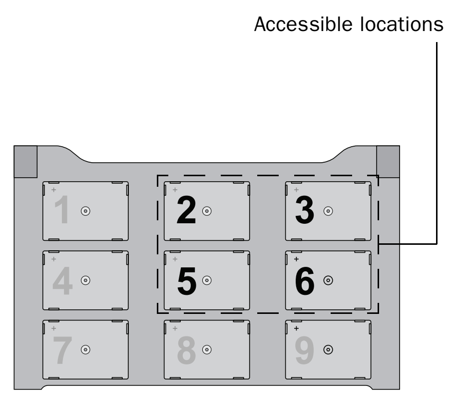
Pipette head	Microplate well format
Series III 96-channel head with disposable tips	96, 384, or 1536
Series III 384-channel head with disposable tips	96, 384 or 1536

Accessible deck locations

The deck locations you can access depends on the pipette head channels you select. The following table shows the channel selections and corresponding deck location limits. Use this information when you set up labware on the Bravo deck.

Pipette channel selection	Accessible Bravo deck locations
	
	
	

Pipette channel selection	Accessible Bravo deck locations
	
	
	

<table>
<tr><th>Pipette channel selection</th><th>Accessible Bravo deck locations</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>
</td><td>
</td></tr>
</table>

Setting the Set Head Mode task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Set Head Mode task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Set Head Mode task in a protocol

Set Head Mode task defined

The Set Head Mode (Bravo) task specifies the channels (or barrels) in the pipette head to be used for pipetting. You can select an m × n array of channels (barrels) for one of the following configurations:

- ❑ All of the pipette channels
- ❑ The first or last full column or row of pipette channels
- ❑ Multiple full columns or rows of pipette channels
- ❑ The first or last partial column or row of pipette channels
- ❑ Multiple partial columns or rows of pipette channels
- ❑ A single pipette barrel at the corner of the pipette head

This task is available for Bravo sub-processes and should only be used if the Series III pipette head is installed.

Before you start

Make sure:

- ❑ You have installed the correct pipette head on the Bravo. See "Pipette head requirements" on page 38 and the *Bravo User Guide*.
- ❑ If you are going to use partial rows or columns of channels (barrels) on the pipette head, retract the tip box stripper pins on the pipette head. See the *Bravo User Guide* for this procedure.
- ❑ The correct Bravo device file is open in the VWorks4 window.
- ❑ The profile you selected shows the correct head type, tip type, and miscellaneous settings.
- ❑ All the teachpoints have been added and verified.

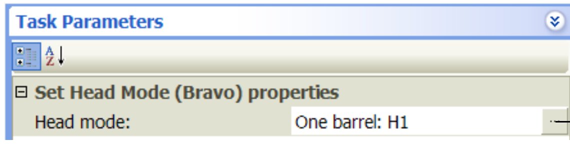
Procedure

To set the Set Head Mode task parameters:

1. In the **Main Protocol** area, at the desired point in the Bravo sub-process, add the Set Head Mode (Bravo) task.

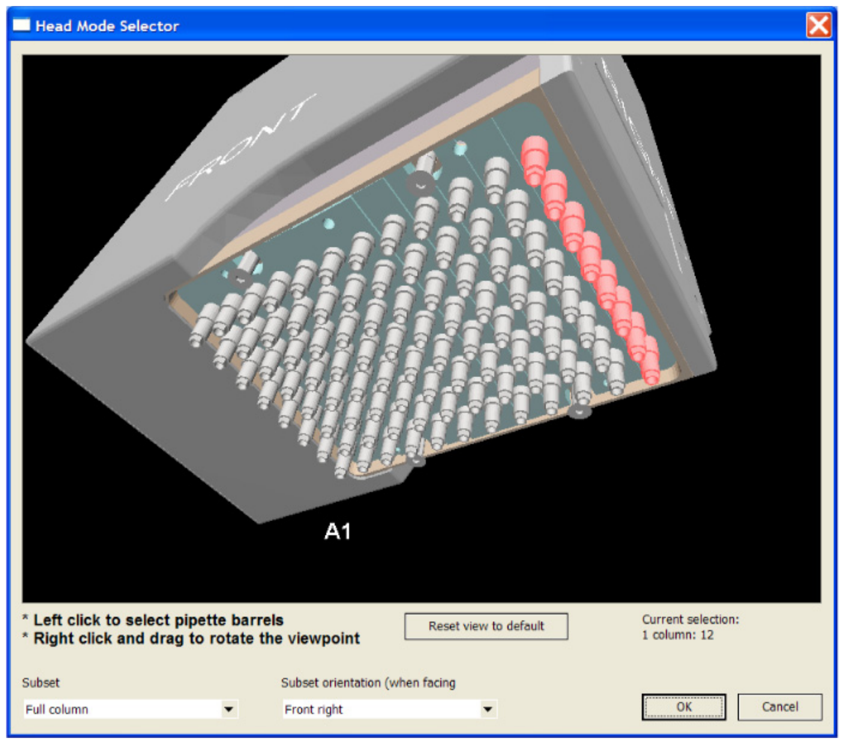
   !! IMPORTANT !! The Set Head Mode task should precede the Serial Dilution task. If you plan to change tips during the serial dilution process, add the Set Head Mode task before the Tips On task.

2. In the Task Parameters area, click **Head mode,** and then click ….



Click to display the Head Mode Selector dialog box.

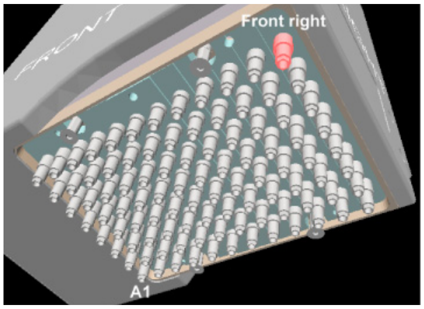
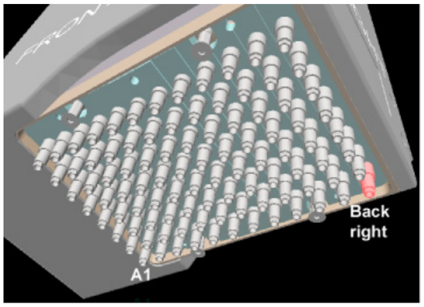
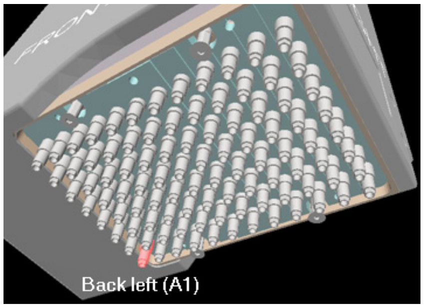
The Head Mode Selector dialog box opens.

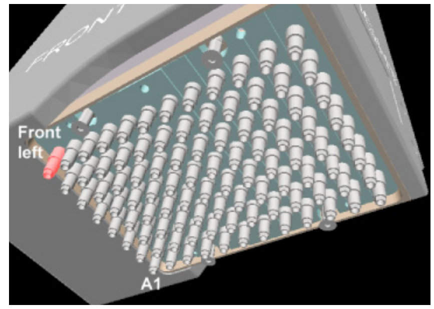


3. In the **Subset** list, select one of the following:

Subset selection	Description
All barrels	Uses all of the pipette channels.
Full column	Uses one or more full columns of pipette channels, starting from the right-most or left-most column.
Full row	Uses one or more full rows of pipette channels, starting from the first row or the last row.
Partial row/column	Uses part of the selected columns or rows.
Single barrel	Uses the single barrel at one of the four corners of the pipette head.

4. In the **Subset orientation** list, select one of the following:

Subset selection	Description
Front right	Uses pipette channels that contain the single channel in the front right corner. 
Back right	Uses pipette channels that contain the single channel in the back right corner. 
Back left	Uses pipette channels that contain the A1 channel (single channel in the back left corner). 

Subset selection	Description
Front left	Uses pipette channels that contain the single channel in the front left corner. 

5. When you are finished, click **OK** to save the selection.

Tracking pipette tip usage

This section contains the following sections:

- "Requirements" on page 48
- "Turning on tip tracking" on page 49
- "Setting the tip box status" on page 51
- "Reusing pipette tips" on page 55

Requirements

About this topic

New options in protocol tasks permit the tracking of pipette tip usage during and across protocol runs. This topic explains the requirements that must be met to use the tip-tracking feature.

Enabling pipette-tip tracking

To enable VWorks4 to track pipette tip usage automatically, make sure you:

- ❑ Turn on the tip-tracking feature. See “Turning on tip tracking” on page 49.
- ❑ Set the starting status of the tip boxes. See “Procedure” on page 51.

Reusing pipette tips

If you want to reuse pipette tips for a portion of the protocol run, turn off the Mark tips as used option in the Tips Off task. For details, see “Reusing pipette tips” on page 55.

Tracking usage across different protocols

If you want to track usage across different protocols, make sure:

- ❑ The protocols reference the same device file.
- ❑ The tip boxes are at the same physical deck locations.
- ❑ In the software, the tip boxes are configured at the same locations across the protocols.

Note: When a set of tip boxes are designated as a process plate, tip usage is tracked during the protocol run. At the end of a run, the software resets the tip box to the original state. The software assumes that in each subsequent run, you will load tip boxes in the original state before the run.

Turning on tip tracking

About this topic

This topic explain how you can turn on trip tracking for a protocol run.

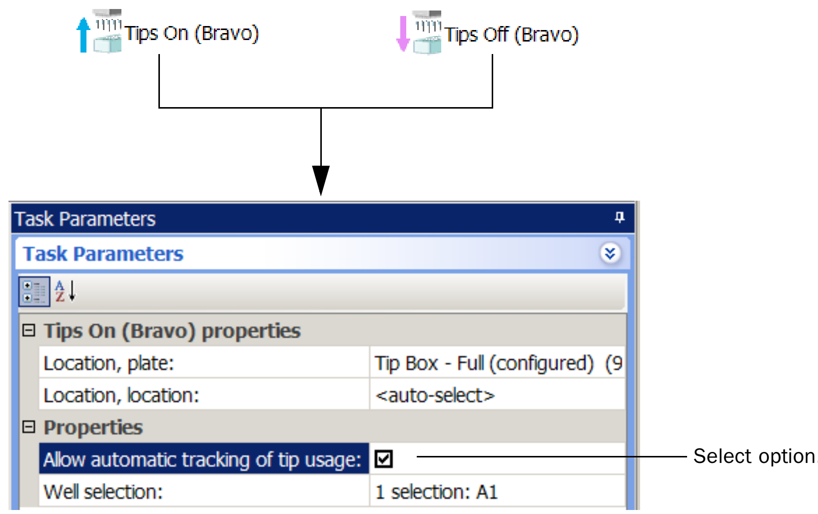
Requirements

To track pipette tip usage during a run, you must turn on the tip-tracking feature in:

- Tips On and Tips Off tasks
- Serial Dilution task (if used in the protocol)

Tips On and Tips Off tasks

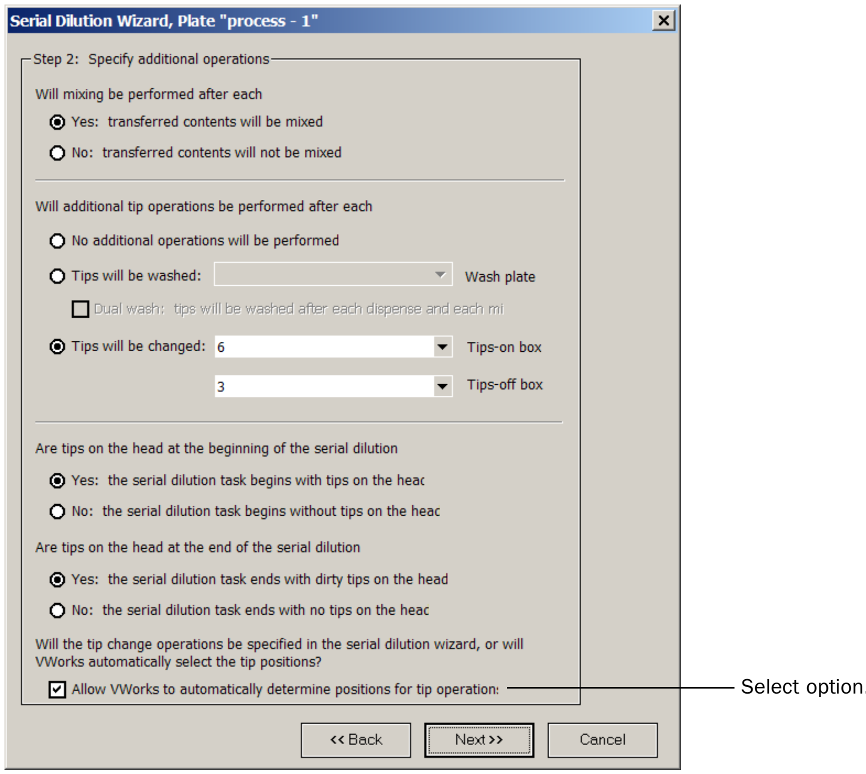
If you added the Tips On and Tips Off tasks in the protocol, in the **Task Parameters** area of both tasks, select **Allow automatic tracking of tip usage.**



Serial dilution task

If you want to turn on tip tracking in a serial dilution protocol, you must first turn on the tracking options in the Tips On and Tips Off tasks.

In a serial dilution protocol, in the **Serial Dilution** task, in step 2 of the wizard, select **Allow VWorks to automatically determine positions for tip operation.** To add the Serial Dilution task, see the *VWorks4 User Guide*.



Setting the tip box status

About this topic

If you turned on automatic pipette-tip tracking, the software allows you to set the current status of tip boxes. Knowing the status of the tip boxes enables the software to accurately track usage during the run within the same protocol and across multiple protocols.

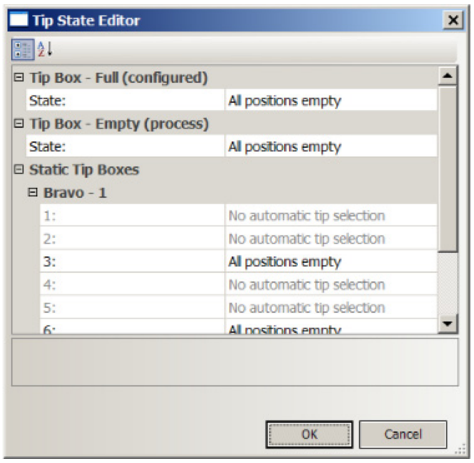
This topic explains how to set up the tip box status so that the software can track the pipette tip usage during and across protocols.

Procedure

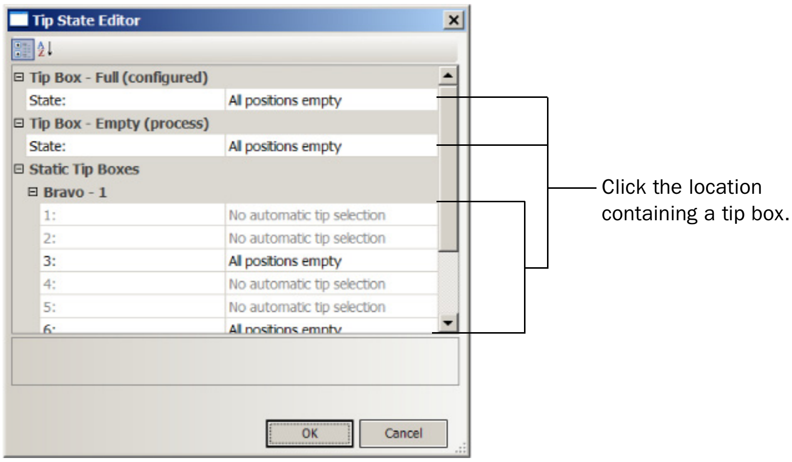
To set the tip box status:

1. On the **Options** menu, click **Automatic Tip State Editor.** The Tip State Editor dialog box opens and displays the tip boxes.

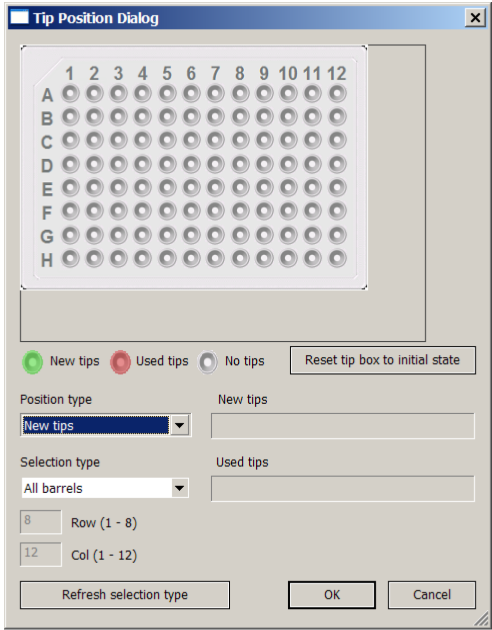
   The tip boxes that are process plates or configured labware are listed by name (specified by the Plate Name property). Tip boxes that are static are listed by deck location. (For a definition of process plates, configured labware, and static labware, see "Creating a protocol" on page 7.)



2. Click one of the tip boxes.



The Tip Position dialog box opens.



3. In the **Position type** list, select the type of tips you are specifying in the tip box graphic.

Selection	Description
New tips	Select if you are specifying new tips in the tip box graphic. For example, if this is a Tips On tip box, you might want to select New tips to indicate the number and position of new pipette tips in the box.
Used tips	Select if you are specifying used tips in the tip box graphic. For example, if this is a Tips Off tip box, you might want to select Used tips to indicate the number and position of used pipette tips in the box.

4. In the **Selection type** list, select how you want to highlight the tip-holding spaces in the graphic:

Selection	Description
All barrels	Select to highlight all of the tip-holding spaces in the graphic in one click.
Full column	Select to highlight one column in one click.
Full row	Select to highlight one row in one click.

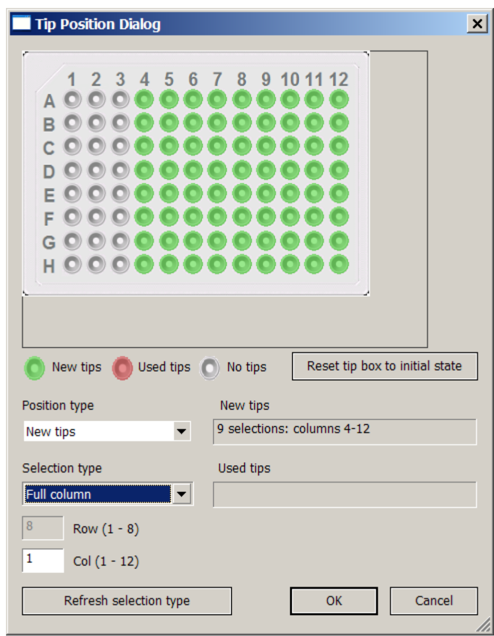
Selection	Description
Partial row/column	Select to highlight a specified number of tip-holding spaces in a column or row in one click.
Single barrel	Select to highlight a single tip-holding space in one click.

5. *Partial row/column selection type only.* In the **Row** and **Column** boxes, type the number of tip-holding spaces you want to highlight in one click.

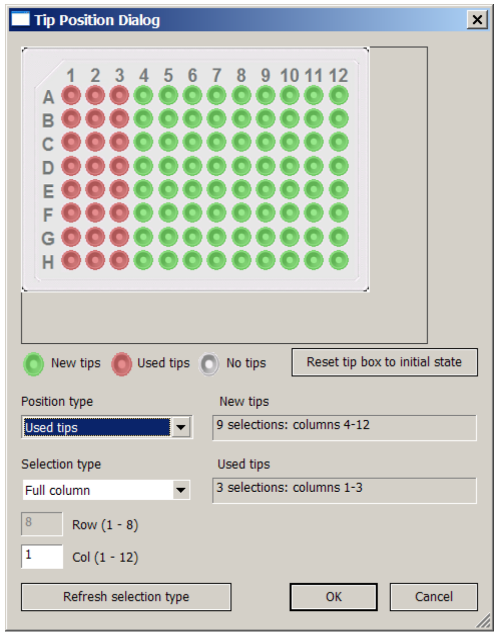
   For example, if you want to select a 4 × 3 array of tip-holding spaces in one click, type 4 in the Row box and 3 in the Column box.

6. In the graphic area, select the tip-holding spaces that you want to mark as containing new or used pipette tips. New pipette tips are green. Used pipette tips are red. Unused tip-holding spaces are gray.

   In the following example, the graphic shows a tip box containing new tips in columns 4 through 12. Columns 1 through 3 are empty.



In the following example, the graphic shows a tip box containing used tips in columns 1 through 3. Columns 4 through 12 contain new tips.



7. When you are finished, click **OK** to save the changes.
8. In the Tip State Editor dialog box, click **OK** to save the changes, and then click the **Close** button (X).
9. Save the protocol.

During the simulation or protocol run, the software uses the starting status of the tip boxes. The software counts the number of tips that are used during the run and stores the information. If you use the same protocol later, the software remembers the tip count and will display the current status in the Tip Position dialog box. You can verify the information before starting the run.

The software can also remember the tip count when you open a different protocol. See “Tracking usage across different protocols” on page 48 for instructions.

Reusing pipette tips

About this topic

When you turn on tip tracking, the software counts the number of tips used during the protocol run. The tips that have been used once are marked as used so that they cannot be picked up and reused.

You have the option of specifying that the tips be reused. For example, in the first part of the protocol, you can reuse the pipette tips that you used to add buffer solution to a microplate.

This topic explains how to specify the reuse of pipette tips during a protocol.

Procedure

To specify that you want to reuse pipette tips in the software:

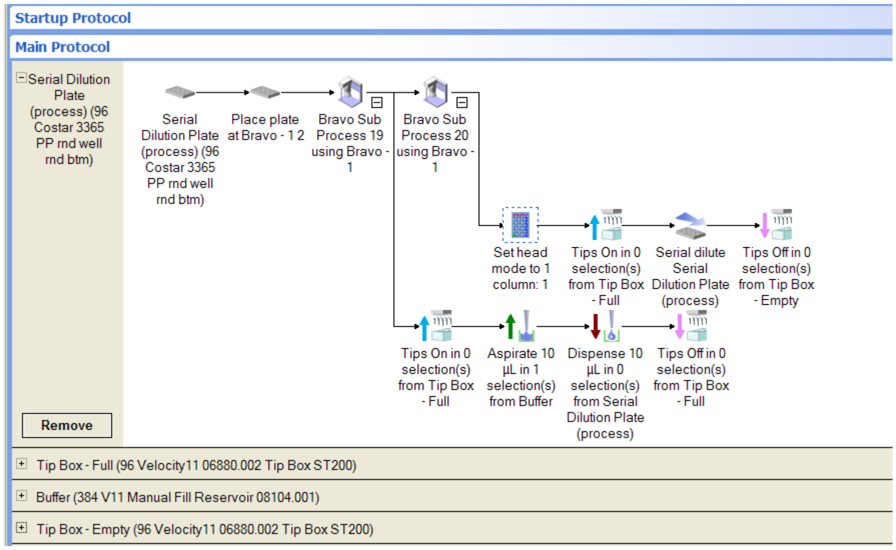
1. In the protocol, locate the Tips Off task where you expect to reuse the tips that are removed.
2. Select the **Tips Off** task.
3. In the **Task Parameters** area, clear the **Mark tips as used** check box.

   During the protocol run, when the tips are removed, the software does not mark the pipette tips as used. During the next Tips On task, the same tips can be reused.

Example

The following example protocol demonstrates how to use the Mark as used option to specify whether pipette tips can be reused. The protocol is set up as follows:

- The serial dilution microplate is at deck location 2.
- A reservoir containing buffer solution is at deck location 7.
- A full tip box is at deck location 6.
- An empty tip box (for tip removal) is at deck location 9.
- In the first Bravo sub-process, the automatic trip-tracking feature is turned off. In addition, in the Tips Off task, the Mark as used option is not selected.
- In the second Bravo sub-process, the automatic tip-tracking feature is turned on. In the Tips Off task, the Mark as used option is selected.



The protocol performs the following tasks:

1. The software acknowledges the presence of the Serial Dilution Plate at deck location 2.
2. An entire box of pipette tips are installed on the pipette head at deck location 6.
3. Buffer solution is aspirated from the reservoir at deck location 7 and dispensed into the Serial Dilution Plate at deck location 2.
4. Pipette tips are removed at deck location 6.
5. The first-available column of pipette tips are installed on the pipette head at deck location 6. These are the pipette tips that were used to add the buffer solution.
6. Serial dilution is performed.
7. The column of pipette tips are removed at deck location 9 and are marked as used.
8. Steps 5 through 8 are repeated until the serial dilution is finished.

Automating labware stacking on the Bravo deck

This section contains the following sections:

- "About the stacking tasks" on page 58
- "Selecting stacking options in the profile" on page 59
- "Specifying maximum stack height" on page 60
- "Adding the Scanstack task" on page 62
- "Adding the Downstack or Upstack task" on page 63

About the stacking tasks

About this topic

You can use various VWorks4 tasks to automate the stacking of labware on the Bravo deck. This topic describes the tasks required to automate labware stacking on the Bravo Platform.

Stacking task descriptions

Stacking is the process of placing a number of labware at a deck location in an orderly pile. A stack must consist of labware that have the same labware definition.

You use stacking to:

- ❑ Process multiple labware without the use of a labware-handling device such as the BenchCel Microplate Handling Workstation. A stack on the Bravo deck is equivalent to a stack coming from a BenchCel Workstation.
- ❑ Run a protocol without having to manually swap labware. Instead of manually replacing processed labware with a new labware between runs, the Bravo robot can automatically process every labware in the stack in a single run without pausing.
- ❑ Reduce space usage on the Bravo deck. Stacking labware at one deck location frees up deck locations for accessories or other labware.

You can use the following tasks to automate the stacking processes:

Task	Description
Scanstack (BuiltIn)	Scans a specified deck location to confirm the presence or absence of labware and determine the height of the stack.
Downstack (BuiltIn)	Moves a microplate from a specified deck or stack location.
Upstack (BuiltIn)	Moves labware to a specified deck or stack location.

Using the tasks

To use the stacking tasks, you need to:

- ❑ Select the desired stacking options in the profile.
- ❑ Specify the maximum height allowed at the deck or stack location.
- ❑ Add a Scanstack task for each Upstack and Downstack task.
- ❑ Add Upstack and Downstack tasks to the protocol.

Selecting stacking options in the profile

About this topic

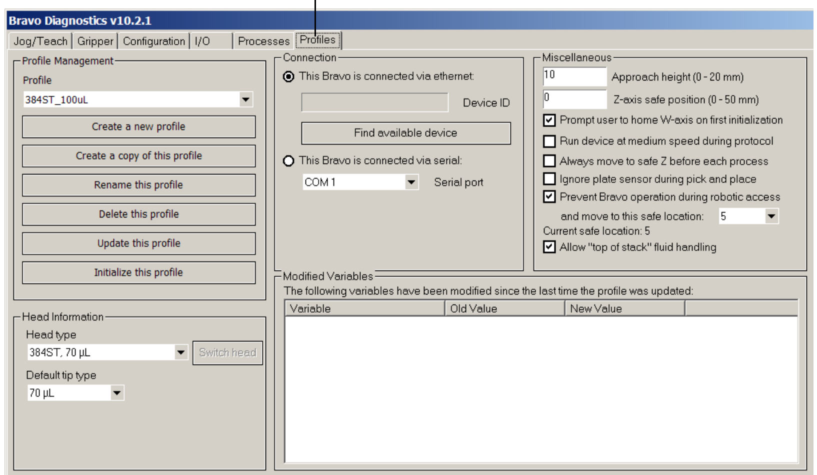
This topic describes the available stacking options you can select in the profile.

Selecting stacking options

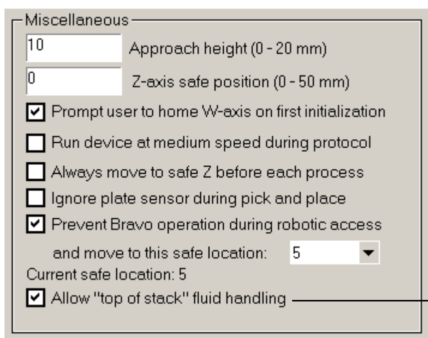
To select the desired stacking option in the profile:

1. Open **Bravo Diagnostics.** For instructions, see the *Bravo User Guide*.
2. Click the **Profiles** tab.

Click to display the Profiles tab.



3. In the **Miscellaneous** area, select the desired stacking options:



Select the desired stacking option.

Option	Description
Allow “top of stack” fluid handling	Permits fluid handling tasks to be performed in the labware at the top of a specified stack.

4. Click **Update this profile** to save the changes.
5. Click **Initialize this profile.**

Specifying maximum stack height

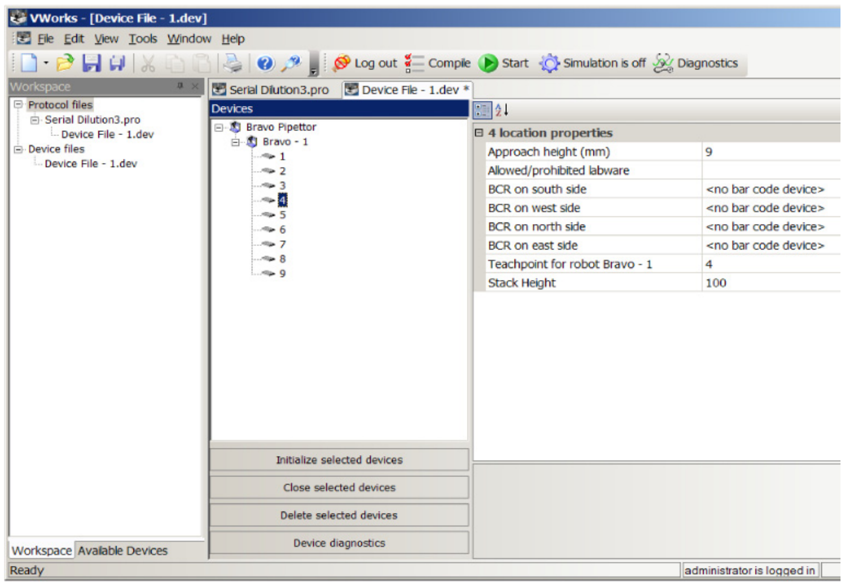
About this topic

This topic explains how to specify the maximum height allowed at each deck location that will host stacking tasks.

Procedure

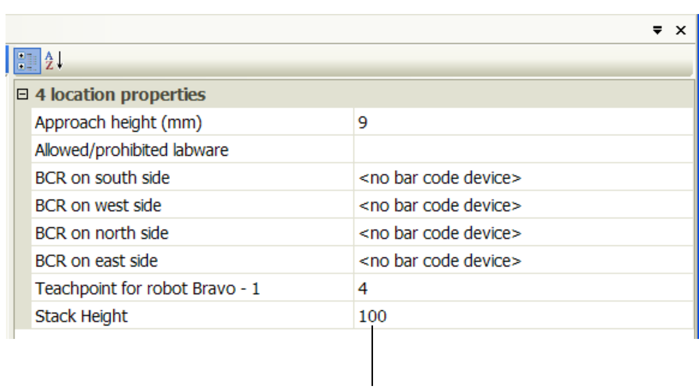
To set the maximum height allowed at each stack:

1. Open the device file.
2. Select the deck location at which you will perform stacking tasks.

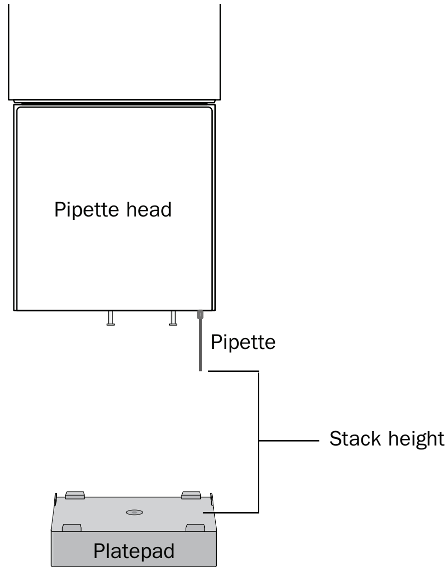


3. In the **Location Properties** area, type the vertical distance from the top surface of the platepad to the bottom of the pipette tips in the **Stack Height** box. The software uses the Stack Height parameter value to determine the maximum number of labware that can be stacked at the location.

   Velocity11 recommends that you set the Stack Height at 100 mm or less to accommodate various pipette heads and tip types.



Type the maximum height, in millimeters.



The following table shows possible values you can specify. The default value is 100 mm, the physical distance between the top of the platepad and the bottom of fixed-tip pipettes.

Pipette type	**Max. height above standard platepad***
Fixed-tip	100 mm (default value)
200-µL	110 mm
30-µL	135 mm

* Standard platepads are 30 mm tall.

4. Save the device file.

Adding the Scanstack task

About this topic

To automate labware stacking tasks, you must add Scanstack tasks in the Startup Protocol. The Scanstack task scans a specified deck location to confirm the presence or absence of labware and determine the height of the stack.

This topic explains how to add the Scanstack task and set the task parameters. For information about the Startup Protocol, see the *VWorks4 User Guide*.

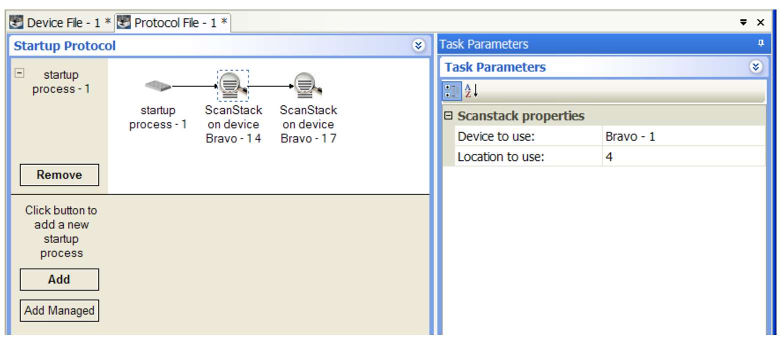
Before you start

You must add one Scanstack task for every Upstack or Downstack task. Before you add the Scanstack tasks in the Startup Protocol, determine the number of stacking tasks you will be adding in the Main Protocol.

Procedure

To add a Scanstack task in the Startup Protocol:

1. In the VWorks4 software, create a new protocol or open an existing protocol.
2. In the **Startup Protocol** area, add the correct number of Scanstack (BuiltIn) tasks.
3. In the **Task Parameters** area, set the following properties:



Property	Description
Device to use	Select the Bravo Platform on which the stacking will occur.
Location to use	Select the deck location to be scanned. If the deck location will be used for a Downstack task, the robot will scan the stack to confirm the presence of the stack and determine its height. If the deck location will be used for an Upstack task, the robot will scan the location to confirm that it is empty and ready to receive labware.

Adding the Downstack or Upstack task

About this topic

The Downstack task moves labware from a specified deck location. The Upstack task moves labware to a specified deck location.

This topic describes how to add the Downstack and Upstack tasks and set the parameters.

Before you start

You must add a Scanstack task for every Downstack or Upstack task in the Main Protocol. See "Adding the Scanstack task" on page 62.

Procedure

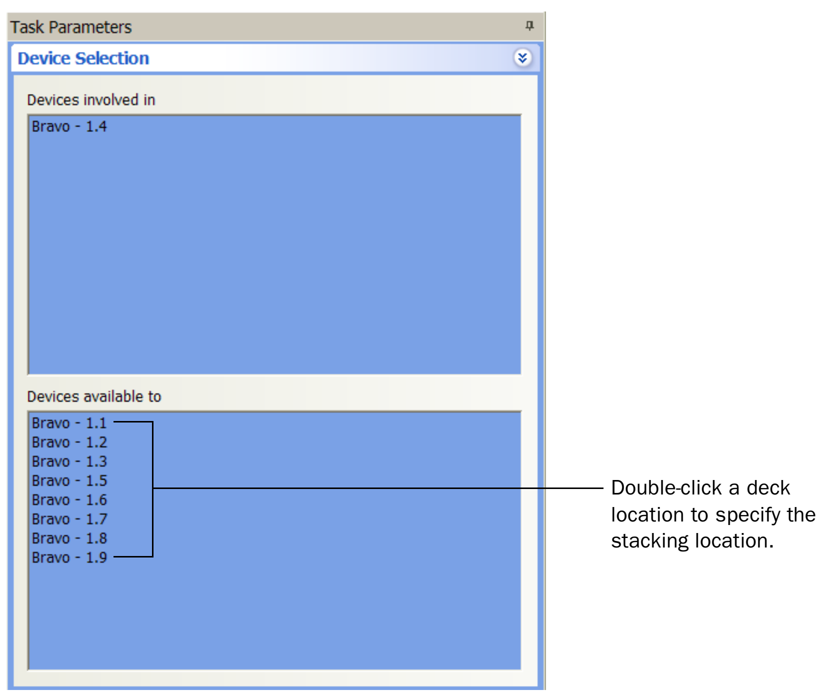
To add a Down Stack or Upstack task in the Main Protocol:

1. In the **Main Protocol** area, add a Downstack (BuiltIn) or a Upstack (BuiltIn) task at the desired point in the protocol.

   Note: The Downstack or Upstack task should be outside of a Bravo sub-process.

2. In the **Device Selection** area, specify the deck location from which the robot should pick up (for downstacking) or place (for upstacking) the labware for the stacking task. To do this, double-click the deck location in the **Devices available to** list. The deck location moves to the Devices involved in list.

   !! DAMAGE HAZARD !! Do not upstack labware next to a serial dilution microplate. Doing so can cause the pipette head to collide with the stack of microplates during the serial dilution task.

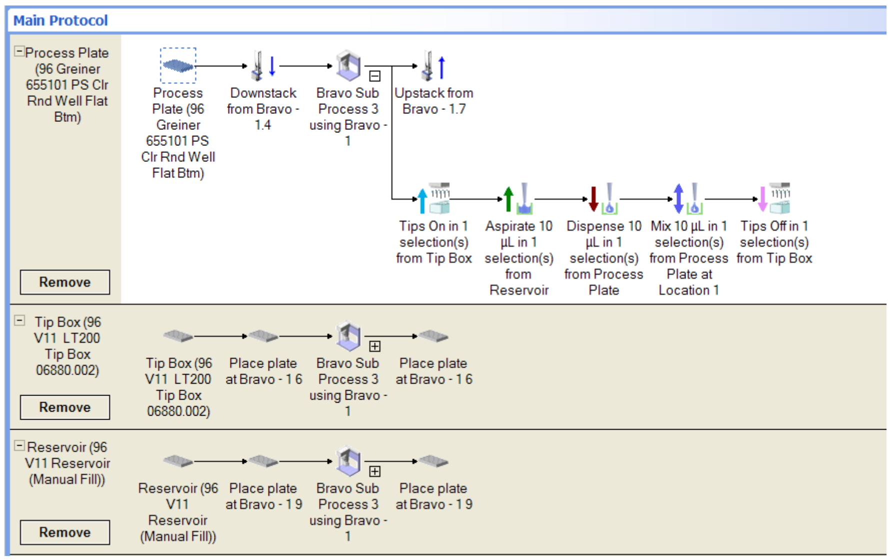
   

3. Add other tasks, such as a Bravo sub-process, to the protocol.

Example

The following example protocol shows the use of the Upstack and Downstack tasks. The protocol is set up as follows:

- ❑ A stack of microplates (Process Plates) is at deck location 4.
- ❑ A tip box (Tip Box) is configured at deck location 6.
- ❑ A Manual Fill Reservoir (Reservoir) is configured at deck location 9.
- ❑ The Startup Protocol (not shown) contains ScanStack tasks for deck locations 4 (where the stack starts) and 9 (where the stack will end up after upstacking).



The protocol performs the following:

1. The robot scans deck location 4 to confirm the presence of the stack and determines the number of microplates. (The ScanStack task in the Startup Protocol, not shown, performs this task.)
2. The robot scans deck location 9 to confirm the absence of labware. (The ScanStack task in the Startup Protocol, not shown, performs this task.)
3. The top-most Process Plate is downstacked from deck location 4 to deck location 1.

   By default, when the <auto-select> location is selected for a process plate task, the process plates are always placed or downstacked at deck location 1. If deck location 1 is occupied, the process plate will be placed or downstacked at the next-available location.
4. Tips are installed on the pipette head at deck location 6.
5. The robot aspirates solution from the Reservoir at deck location 9.
6. The robot dispenses the solution into the Process Plate at deck location 1.
7. The robot mixes the solution in the Process Plate at deck location 1.
8. The pipette tips are removed at the Tip Box at deck location 6.

9. The Process Plate at deck location 1 is upstacked to deck location 9.
10. Steps 1 through 8 is repeated for each Process Plate stacked at deck location 4.

Prompting users for task parameter values

This section contains the following topics:

- "About variables" on page 68
- "Assigning variables and adding code snippets" on page 69
- "Editing variables and code snippets" on page 73\
- "Adding user message prompts" on page 76

About variables

About this topic

This topic explains the use of variables in task parameters.

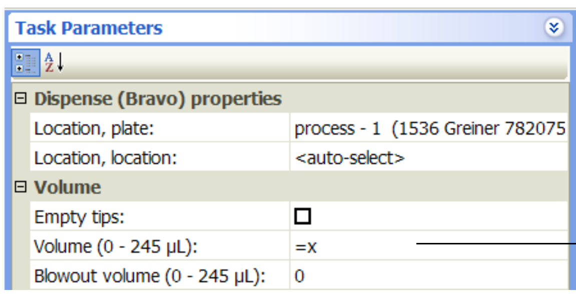
Description

You can use variables to assign user-supplied values to task parameters. For example, during a protocol run, the software will prompt the user for aspirate and dispense volumes. The software will use the user-supplied values during the run.

Instead of writing detailed JavaScript code in the JavaScript task Advanced Settings area, you can:

- ❑ Type a simple variable assignment and a code snippet directly in a field in the Task Parameters area.
- ❑ Type multiple variable assignments and code snippets in the JavaScript task Advanced Settings area.

The following example shows how to assign a variable called *x* to the Dispense Volume parameter in the Task Parameters area.

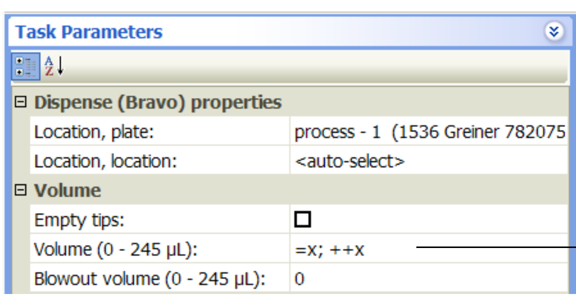


Type a variable assignment in the Task Parameters area.

Assigning the variable in the Task Parameters area is equivalent to providing the following line of code in the Advanced Settings area:

task.Volume = x;

You can also add a snippet of code after the variable assignment. For example, you can assign the Dispense Volume to x, and then increment it by the same amount in each loop during the protocol run.



Type a snippet after the variable assignment.

To prompt the user for the values to use during the protocol run, you add User Message tasks in the Startup Protocol or at the desired points in the Main Protocol. See “Adding user message prompts” on page 76.

Using the correct syntax

When adding a variable and code snippet in the Task Parameters area, use the following syntax:

=x; <code snippet>

where *x* is the variable name and <code snippet> is additional code you want to use for the variable. Use standard JavaScript rules for the variable name and for the code snippet.

Assigning variables and adding code snippets

About this topic

The VWorks4 software allows you to assign variables to almost any task parameter. The way you assign the variable depends on the field input style.

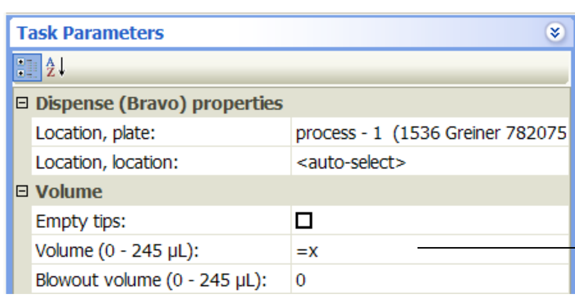
This topic explains how you assign variables in the different input fields and in the JavaScript task Advanced Settings area. Read this topic if you are an administrator or technician who writes protocols.

Note: Parameters that require fixed values do not accept variables. For example, passwords, deck locations, and IP addresses cannot be assigned variables. The software does not allow you to assign variables to some of these parameters. Other parameters, such as Password, will accept the variable itself as a string value rather than a variable assignment.

Text box parameter values

To assign a variable to a parameter whose value is displayed in a text box:

In the parameter value text box, type the variable assignment and optional code snippet. For syntax requirements, see "Using the correct syntax" on page 69.

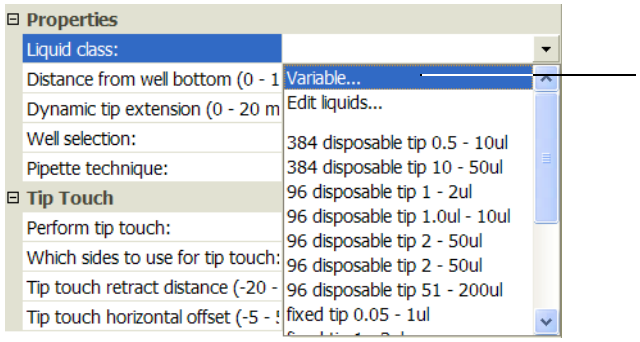


Type a variable assignment and desired code snippet in the text box.

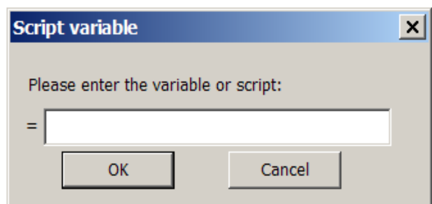
List of parameter values

To assign a variable to a parameter whose value can only be selected from a list:

1. In the list, select **Variable.**



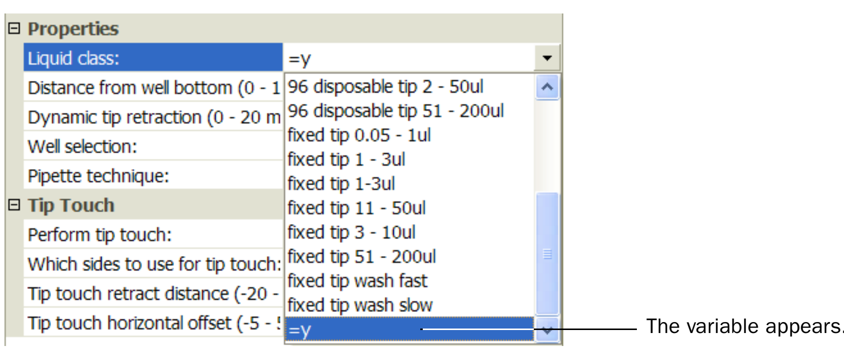
   The Script variable dialog box opens.



2. Type the variable assignment and optional code snippet. For syntax requirements, see “Using the correct syntax” on page 69.

   Note: The = symbol in front of the text box indicates that the software will automatically add the = symbol in front of the variable.

3. Click **OK.** The new variable and optional code snippet appear in the parameter value list.

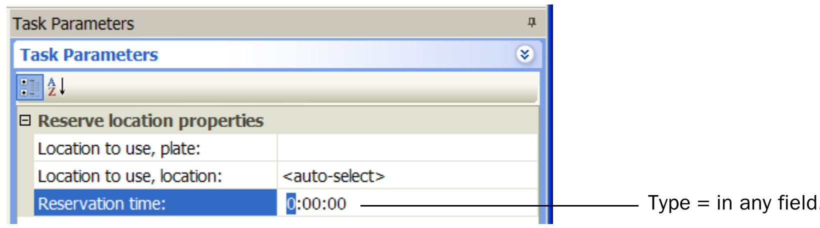


Time format parameter values

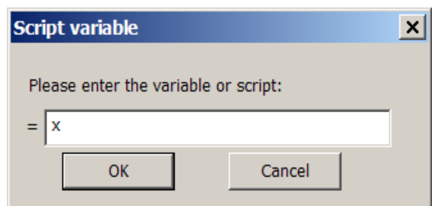
Some tasks, such as the Reserve a location on a device task, have a time parameter in the h:mm:ss format.

To add a variable to a parameter whose value is in the time format (h:mm:ss):

1. In the parameter value box, type **=** in the h, mm, or ss field.



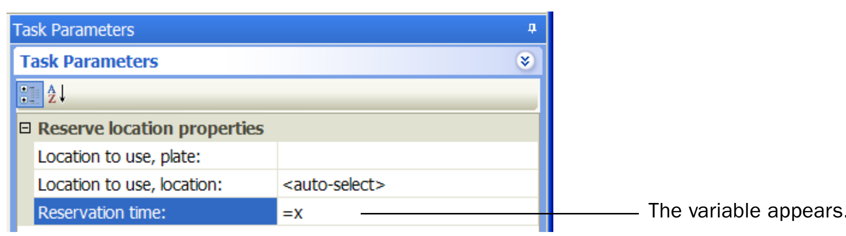
The Script variable dialog box opens.



2. Type the variable assignment and optional code snippet. For syntax requirements, see “Using the correct syntax” on page 69.

   Note: The = symbol in front of the text box indicates that the software will automatically add the = symbol in front of the variable.

3. Click **OK.** The new variable and optional code snippet appear in the parameter field.

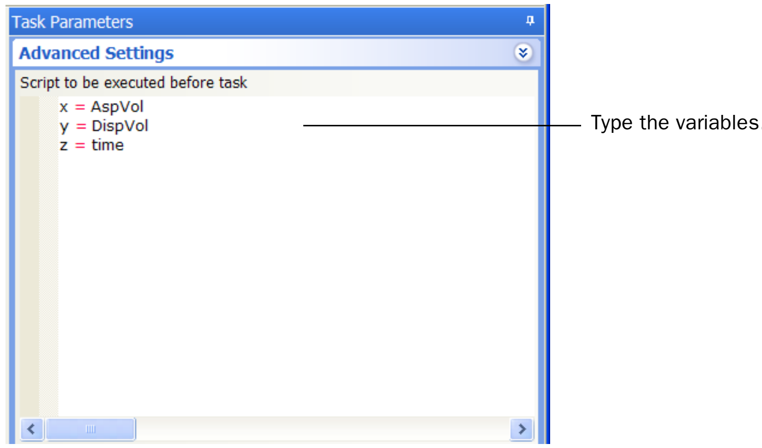


JavaScript task Advanced Settings area

To assign multiple variables to multiple parameters:

1. In the protocol, click **Startup Protocol.**
2. Click **Add Process.** A Startup process appears.
3. Add the  JavaScript (BuiltIn) task.

4. In the **Advanced Settings** area, type the variable assignments.



Editing variables and code snippets

About this topic

You can edit any variable and code snippet that you created for a task parameter. This topic explains how to edit variables that were created in the Task Parameters area.

Editing a variable in a text box

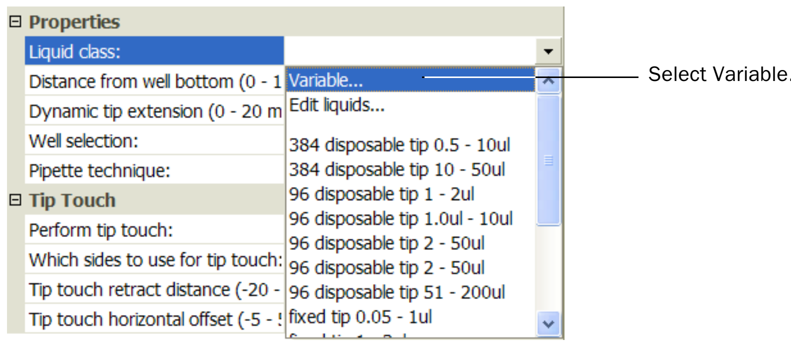
To edit a variable in a task parameter text box:

Type the new variable or code snippet in the text box.

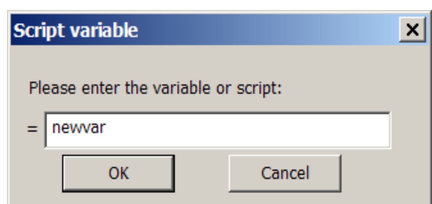
Editing a variable in a list

To edit a variable in a task parameter list:

1. In the list, select **Variable.**



The Script variable dialog box opens.

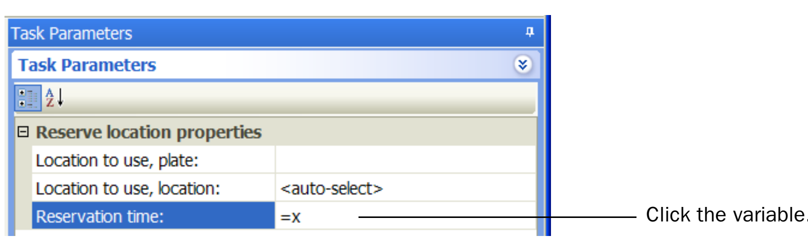


2. Type the new variable assignment and new code snippet. For syntax requirements, see “Using the correct syntax” on page 69.
3. Click **OK.** The new variable and optional code snippet appear in the parameter value list.

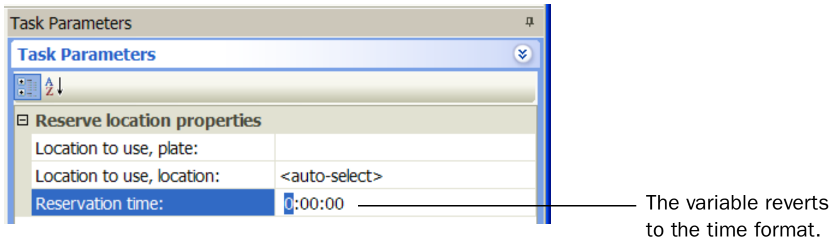
Editing a variable in a time parameter

To edit a variable in a time parameter:

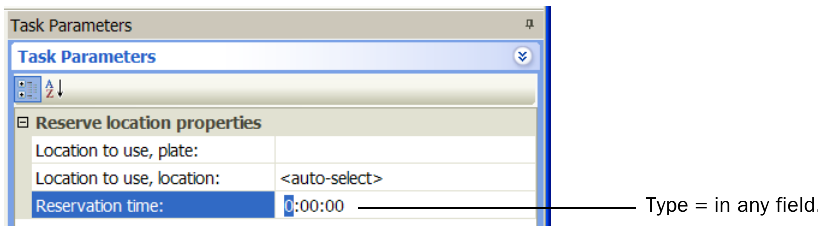
1. Click to select the time field.



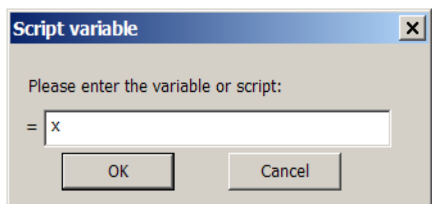
The time field reverts to the h:mm:ss format.



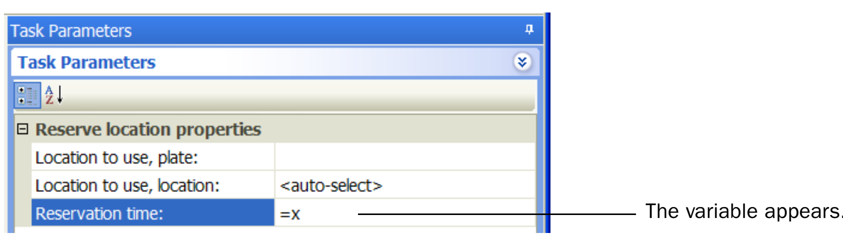
2. Type = in the h, mm, or ss field.



The Script Variable box opens.



3. Type the variable assignment and optional code snippet. For syntax requirements, see “Using the correct syntax” on page 69.
4. Click **OK.** The new variable and optional code snippet appear in the parameter field.

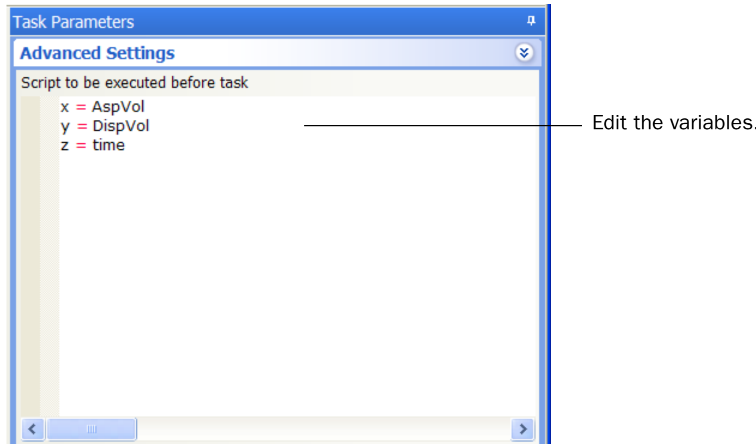


Editing a variable in the Advanced Settings area

To edit variables in the Advanced Settings area:

1. In the protocol, click **Startup Protocol.**
2. Click the JavaScript (BuiltIn) task.

3. In the **Advanced Settings** area, edit the desired variable assignments.



Adding user message prompts

About this topic

After adding variables to the desired task parameters, you can add User Message tasks to prompt users for values at the beginning of the protocol run or at the desired points during the protocol run.

This topic explains how to add User Message tasks to prompt the user for parameter values.

Procedure

To add User Message tasks:

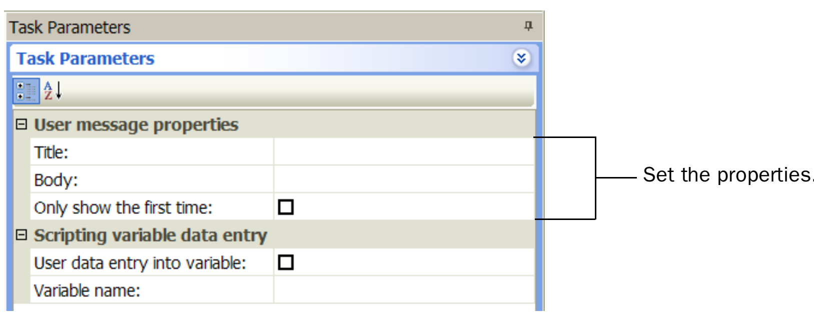
1. In the **Startup Protocol:**

   - Add a User Message (BuiltIn) task for each variable assigned directly in a task parameter input field.

   - Add one User Message (BuiltIn) task for the set of variables you assigned in the Advanced Settings area. If you added variables in the Advanced Settings area, add the User Message task after the JavaScript task in the Startup Protocol.

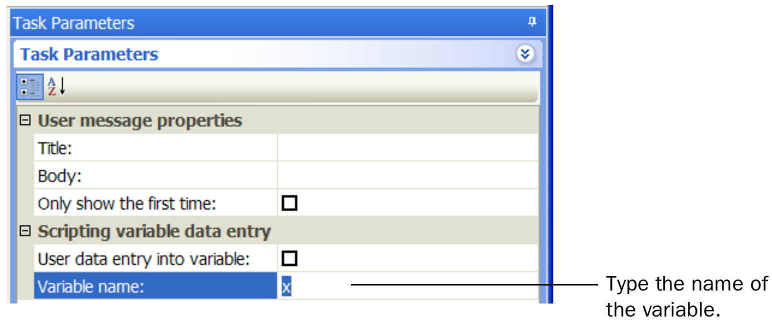
   Alternatively, at the desired points in the **Main Protocol,** add a User Message (BuiltIn) task for each variable assigned directly in a task parameter input field.

2. In the **Task Parameters** area, type values and select the desired options in the **User message properties** table. For a description of the properties, see the *VWorks4 User Guide*.



3. Select **User data entry into variable** and type the name of the variable in the **Variable name** box. During the protocol run, the software will prompt the user for a value and assign it to this variable.

   If you added multiple variables in the Advanced Settings area, do not use the Variable name property. Instead, use the **Body** property to instruct the user to set the variable values in the Advanced Settings area.

Using JavaScript utilities

This section contains the following topics:

❑ "About the JavaScript utilities" on page 80
❑ "ActiveX Wrapper" on page 81
❑ "File Object" on page 82

About the JavaScript utilities

About this topic

This topic lists the JavaScript utilities that are available and explains where you can write the code for the utilities.

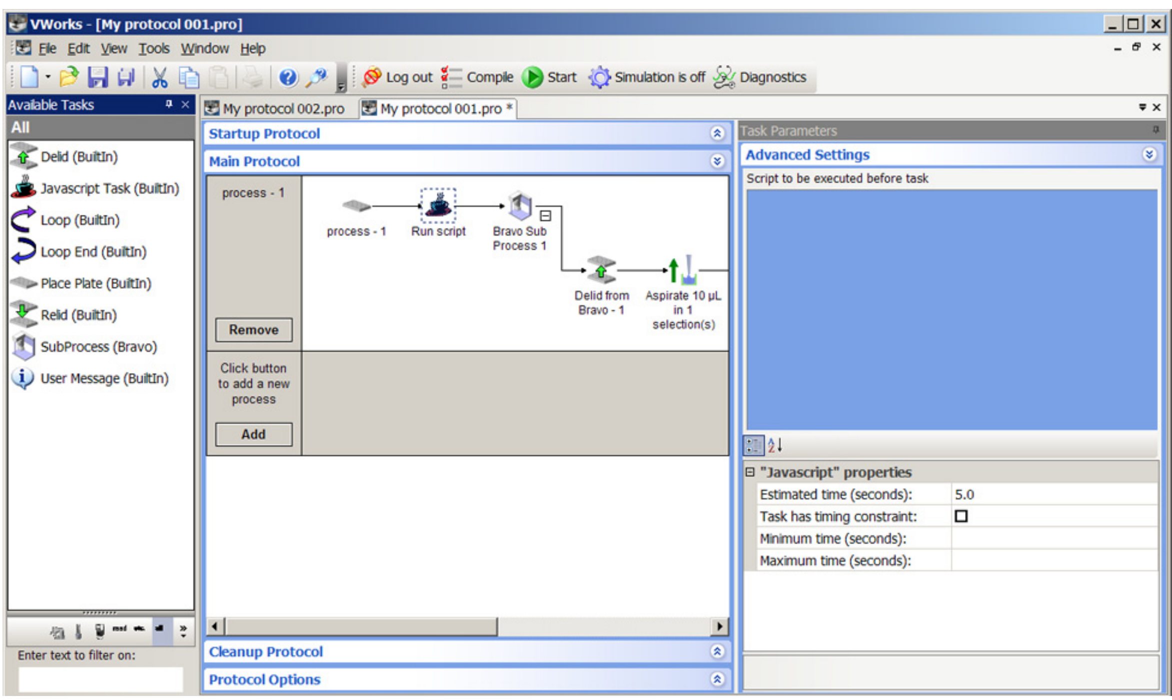
Available utilities

Two JavaScript utilities are available:

- *ActiveX Wrapper*. Allows you to create an ActiveX object and use the associated ActiveX methods.
- *File Object*. Allows you to create a file object to read from and write to a file.

Using the utilities

You can write the JavaScript using these utilities in the VWorks4 **Advanced Settings** area. For instructions on how to add the JavaScript task and display the Advanced Settings area, see the *VWorks4 User Guide*.



ActiveX Wrapper

About this topic

This topic describes the ActiveX Wrapper utility and explains how to use the utility in JavaScript.

About the ActiveX Wrapper utility

ActiveX controls are software components that allow different software products to interact. For example, if you want to use the VWorks4 software to control a third-party device, you can use the device's ActiveX control to invoke the device's operations.

The ActiveX Wrapper utility in the VWorks4 software allows you to use another product's ActiveX control to invoke the product's operations. Make sure you install the product's ActiveX control software before you run the JavaScript.

To use the ActiveX utility, you need to:

1. Create an ActiveX object to reference the ActiveX control.
2. Call the associated ActiveX methods to invoke the ActiveX operations.

Creating an ActiveX object

To create an ActiveX object:

In the **Advanced Settings** area, type the JavaScript code to create an ActiveX object.

For example, if the ActiveX control PROG_ID is PlateLocCtrl.2, you can create the object as follows:

```
var ocx
if( ocx == undefined){
   ocx = new ActiveX( "PLATELOC.PlateLocCtrl.2");
}
```

The `var` statement declares a JavaScript variable. In this example, the variable is `ocx`.

The `if` statement prevents the software from creating the ActiveX multiple times if the script is run repeatedly.

The `ocx = new ActiveX` statement passes the PlateLoc PROG_ID to the ActiveX object generator. Using the ID, the generator calls the CreateInstance API. The resulting ActiveX object is then wrapped in the scripting layer that translates arguments and returns values that are understood by both the PlateLoc and VWorks4.

Calling the ActiveX methods

To call the ActiveX methods:

Call the methods using the following syntax:

```
objectname.method
```

For example, if you want to call the AboutBox() method, you can type the following:

```
ocx.AboutBox()
```

To list the available ActiveX methods, use the following JavaScript statements:

```
for( x in ocx.members)
   print( x)
```

File Object

About this topic

This topic describes the File Object utility and explains how to use it in JavaScript.

About the FileObject utility

The File Object utility allows you to create a file object so that you can read from and write to a file. To use the File Object utility, you need to:

1. Create the file object.
2. Call the desired file object methods:
   - Open
   - Close
   - Read
   - Write
   - IsOpen

!! IMPORTANT !! JavaScript is case-sensitive. Make sure you use the correct upper- and lower-case letters when calling the methods.

Creating a file object

To create a file object:

In the **Advanced Settings** area, type the following JavaScript code:

```
var fileobjectname
if( fileobjectname == undefined){
fileobjectname = new File()
}
```

Note: `fileobjectname` is the name of the file object you want to create.

Calling the Open method

To call the Open method:

Type the following JavaScript code:

```
fileobjectname.Open( “filepath”, 0, 0)
```

`fileobjectname` is the name of the file object you created.

`filepath` (the first argument) is the location of the file you are creating. For example, you can type `c:\fileobjectname.txt`.

`0` (the second argument) specifies how new information will be added to the file. 0 adds new information after the existing information. A non-zero value erases the existing file contents before adding the new information. If you do not specify this argument, the system will use the default value of 0.

`0` (the third argument) specifies how the line endings in binary files will be translated. 0 translates line endings to a carriage return followed by a line feed. 1 does not translate the existing line ending. A non-zero value erases the existing file contents before adding the new information. If you do not specify this argument, the system will use the default value of 0.

Calling the Close method

To call the Close method:

Type the following JavaScript code:

```
fileobjectname.Close()
```

`fileobjectname` is the name of the file object you created.

The Close method closes the file and releases any locks on the file so that other software can access it.

Calling the Read method

To call the Read method:

Type the following JavaScript code:

```
var result = fileobjectname.Read()
```

The `var` statement reads the entire file and stores the file contents as a string in the `result` variable. Although line-by-line reading is not available, you can use built-in JavaScript string methods to parse the file.

If another process is concurrently adding information to the file, later calls to the Read method will read the newly added information.

Calling the Write method

To call the Write method:

Type the following JavaScript code:

```
fileobjectname.Write( “writeoutput\r\n”)
```

`fileobjectname` is the name of the file object you created.

`writeoutput` is the string you want to add to the file.

`\r` adds a linefeed at the end of the string.

`\n` adds a new line at the end of the string.

Calling the IsOpen method

To call the IsOpen method:

Type the following JavaScript code:

```
var open = fileobjectname.IsOpen()
```

The `var` statement checks to see if the file opening call was successful.

Using the VWorks4 ActiveX control

This section contains the following topics:

- "About the VWorks4 ActiveX control" on page 86
- "Methods" on page 87
- "Events" on page 94
- "Enumerated types" on page 97

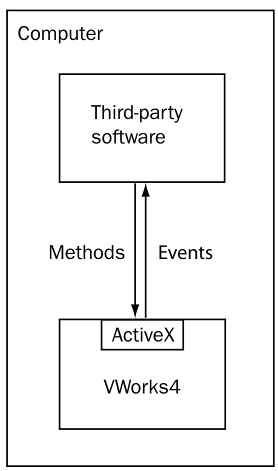
About the VWorks4 ActiveX control

VWorks4 ActiveX control description

The VWorks4 ActiveX control allows you to invoke VWorks4 operations from another software. To use the VWorks4 ActiveX control, you need to know the following:

- *Methods*. Functions that can be called to invoke individual operations
- *Properties*. Variables that are used in methods (for example, speed = fast)
- *Events*. Notifications that methods have completed or resulted in errors
- *Enumerated types*. Constants that are used to indicate the status of a method call.

The following diagram illustrates the use of the VWorks4 ActiveX control. Actions you perform are conducted through ActiveX methods. System responses are relayed back through ActiveX events.



Integrating the VWorks ActiveX control

When integrating the VWorks4 ActiveX control in a third-party lab software:

1. Install the VWorks4 ActiveX control. To install the VWorks4 ActiveX control:
   a. Insert the VWorks4 Software CD into the controlling computer CD-ROM drive.
   b. In the CD folder, double-click VWorks4 Installer.exe.
   c. Follow the directions in the installation wizard window.
2. Open the Command Prompt window and type `VWorks/register` to register the application program interface.
3. Refer to the description of the Methods and Properties in this section.

Methods

AbortProtocol

Description

Aborts the protocol run that is in progress.

Parameters

None

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode = oVWorks4COM.AbortProtocol();
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode = oVWorks4COM.AbortProtocol()
```

CloseProtocol

Description

Closes the specified protocol file.

Parameters

Name	Type	Description
protocol	BSTR	The protocol file path.

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97. *Note:* returnCode is RETURN_SUCCESS if the file closed successfully.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode = oVWorksCOM.CloseProtocol(“Yourprotocol.pro”);
```

Visual Basic .NET example

```
Dim vwRetCode As VWorks4Lib.V11ReturnCode
vwRetCode = oVWorksCOM.CloseProtocol(“Yourprotocol.pro”)
```

CompileProtocol

Description

Compiles the protocol and is used with the LogMessage event.

Parameters

Name	Type	Description
protocol	BSTR	The protocol file path.
errorCount	*LONG	The number of errors found.
warningCount	*LONG	The number of warnings found.

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
LONG errCount, wrnCount;
retCode = oVWorks4COM.CompileProtocol (“c:\myprotocol.pro”,
&errCount, &wrnCount);
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
Dim errCount, wrnCount as Long
retCode = oVWorks4COM.CompileProtocol (“c:\myprotocol.pro”,
errCount, wrnCount)
```

GetSimulationMode

Description

Gets the simulation mode state.

Parameters

None

Returns

Name	Type	Description
mode	VARIANT_BOOL	The value that indicates the simulation state: ❑ True = The simulation mode is on. ❑ False = The simulation mode is off.

Visual C++ example

```
VARIANT_BOOL bSimMode;
bSimMode= oVWorksCOM.GetSimulationMode();
```

Visual Basic .NET example

```
Dim bSimMode as Boolean
bSimMode= oVWorksCOM.GetSimulationMode()
```

LoadProtocol

Description

Loads the protocol for a run.

Parameters

Name	Type	Description
protocol	BSTR	The protocol file path.

Returns

Name	Type	Description
returnCode	V11ReturnCode	See "Enumerated types" on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode=oVWorks4COM.CompileProtocol("c:\myprotocol.pro");
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode=oVWorks4COM.CompileProtocol ("c:\myprotocol.pro")
```

Login

Description

Logs into VWorks using the provided user name and password.

Parameters

Name	Type	Description
userName	BSTR	The user name.
password	BSTR	The password.

Returns

Name	Type	Description
loginResult	V11LoginResult	See "Enumerated types" on page 97.

Visual C++ example

```
VWorks4Lib.V11LoginResult retCode;
loginResult= oVWorksCOM.Login("user1","mypassword!");
```

Visual Basic .NET example

```
Dim loginResult as VWorks4Lib.V11LoginResult
loginResult= oVWorksCOM.Login("user1","mypassword!")
```

Logout

Description

Logs out the current user session.

Parameters

None

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode = oVWorksCOM.Logout();
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode = oVWorksCOM.Logout()
```

PauseProtocol

Description

Pauses the protocol run that is in progress. The tasks that are in progress will be finished. No new tasks will be started.

Parameters

None

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode = oVWorksCOM.PauseProtocol();
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode = oVWorksCOM.PauseProtocol()
```

ResumeProtocol

Description

Resumes the protocol run.

Parameters

None

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode = oVWorksCOM.ResumeProtocol ();
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode = oVWorksCOM.ResumeProtocol ()
```

RunProtocol

Description

Starts the protocol run.

Parameters

Name	Type	Description
protocol	BSTR	The protocol file path.
runCount	LONG	The number of times to run the protocol.

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode=oVWorks4COM.RunProtocol (“c:\myprotocol.pro”,2);
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode=oVWorks4COM.RunProtocol (“c:\myprotocol.pro”,2)
```

SetSimulationMode

Description

Sets the simulation mode.

Parameters

Name	Type	Description
mode	VARIANT_BOOL	The value that sets the simulation state: ❑ True = Turns on the simulation mode. ❑ False = Turns off the simulation mode.

Returns

Name	Type	Description
returnCode	V11ReturnCode	See "Enumerated types" on page 97. *Note:* returnCode is always RETURN_SUCCESS.

Visual C++ example

```
oVWorksCOM.SetSimulationMode(VARIANT_TRUE);
oVWorksCOM.SetSimulationMode(VARIANT_FALSE);
```

Visual Basic .NET example

```
oVWorksCOM.SetSimulationMode(True)
oVWorksCOM.SetSimulationMode(False)
```

ShowVWorks

Description

Displays or hides the VWorks window.

Parameters

Name	Type	Description
showOrHide	VARIANT_BOOL	The value that displays or hides the window: ❑ TRUE = Display the window. ❑ FALSE = Hide the window.

Returns

Name	Type	Description
returnCode	V11ReturnCode	See “Enumerated types” on page 97.

Visual C++ example

```
VWorks4Lib.V11ReturnCode retCode;
retCode=oVWorks4COM.ShowVWorks (VARIANT_TRUE);
```

Visual Basic .NET example

```
Dim retCode as VWorks4Lib.V11ReturnCode
retCode=oVWorks4COM. ShowVWorks (True)
```

Events

LogMessage

Description

The LogMessage event occurs every time a message is posted to the log.

Parameters

Name	Type	Description
session	LONG	The session ID.
logClass	LONG	The type of log event: Error, Warning, or Event.
timeStamp	BSTR	The time at which the error occurred.
device	BSTR	The device name. An empty string is permitted.
location	BSTR	The location. An empty string is permitted.
process	BSTR	The process plate name. An empty string is permitted.
task	BSTR	The task name. An empty string is permitted.
fileName	BSTR	The protocol file or device file name. An empty string is permitted.
message	BSTR	The the error message text.

Returns

None

ProtocolComplete

Description

The ProtocolComplete event occurs after startup, cleanup, and main protocols are finished.

Parameters

Name	Type	Description
session	LONG	The session ID.
protocol	BSTR	The protocol file path.
protocol_type	BSTR	The type of protocol: Startup, Main, or Cleanup.

Returns

None

ProtocolAborted

Description

The ProtocolAborted event occurs when the operator or automation client aborts the protocol run.

Parameters

Name	Type	Description
session	LONG	The session ID.
protocol	BSTR	The protocol file path.
protocol_type	BSTR	The type of protocol: Startup, Main, or Cleanup.

Returns

None

RecoverableError

Description

The RecoverableError event occurs whenever an error is displayed and expects the operator to abort, retry, or ignore the error.

Parameters

Name	Type	Description
session	LONG	The session ID.
device	BSTR	The device name. An empty string is permitted.
location	BSTR	The location. An empty string is permitted.
description	BSTR	The description of the error.

Returns

Name	Type	Description
actionToTake	*LONG	The value that indicates the action to take: ❑ 0 = Abort ❑ 1 = Retry ❑ 2 = Ignore

Name	Type	Description
vworksHandlesError	*VARIANT_BOOL	Value values are: ❑ TRUE = Allows the VWorks4 software to handle the error. the VWorks4 software will not display the error message. ❑ FALSE = Prevents the VWorks4 software from handling the error.

UnrecoverableError

Description

The UnrecoverableError event occurs when an error is displayed and does not expect the operator to respond with a decision.

Parameters

Name	Type	Description
session	LONG	The session ID.
description	BSTR	The description of the error.

Returns

None

Enumerated types

V11ReturnCode

Description

Indicates the method call status.

Constants

Name	Value	Description
RETURN_SUCCESS	0	The method was called successfully.
RETURN_BAD_ARGS	1	The method contains bad arguments.
RETURN_FAIL	2	The method call failed.

V11LoginResult

Description

Indicates the login status.

Constants

Name	Value	Description
LOGIN_SUCCESS	0	The login was successful.
LOGIN_FAIL	1	The login failed.
LOGIN_DISABLED	2	The login was disabled.
LOGIN_EXPIRED	3	The login period passed.
LOGIN_TOO_MANY_FAILED_ATTEMPTS	4	Too many login attempts were made and failed.

Contents

Chapter 1. Introduction 1
Who should read this guide 2
What this guide covers 2
What is new in this version 3
Accessing Velocity11 user guides 4
Finding your version numbers 6
Reporting VWorks problems 7
Sending a bug report 8

Chapter 2. VWorks overview 11
Description of VWorks 12
Devices you can use with VWorks 13
Overview of the VWorks user interface 14
Relationships of VWorks components 24

Chapter 3. Preparing for a run 27
Workflow for preparing a run 28
Starting VWorks 29
Logging in to VWorks and changing your password 30
Opening a protocol in VWorks 31
Notification of errors by email 32
Setting log file directories 34
Understanding the protocol 35

Chapter 4. Performing a run 37
Workflow for performing a run 38
Starting a run from VWorks 39
Pausing or stopping a run 41
Monitoring a run 43
Working with the Log toolbars 44
Closing down after a run 46

Chapter 5. Installing VWorks and networking the controlling computer 47
Workflow for installing VWorks and networking the computer 48
Computer and networking requirements 49
Installing VWorks software 50

Choosing between serial and Ethernet connections . . . 51
About Ethernet networking . . . 52
Workflows for Ethernet networking . . . 55
Setting the network card's IP address . . . 56
Connecting Ethernet cables . . . 59
Checking the network card's IP address . . . 60
Installing and starting DHCP server software . . . 61
Turning off Ethernet firewall software . . . 63
Uninstalling VWorks . . . 65

Chapter 6. Creating a protocol . . . 67
About tasks, processes, and protocols . . . 68
About protocol files . . . 71
An example protocol . . . 72
Workflow for creating a protocol . . . 73
Creating a new protocol . . . 74
Setting protocol options . . . 75
Setting the plate properties for a protocol process . . . 77
About setting the number of simultaneous plates . . . 79
Setting up a startup or cleanup protocol process . . . 80
Adding and deleting tasks . . . 81
Compiling protocols . . . 83
Saving protocols . . . 84
Simulating a run . . . 84
Tracking plates using bar codes . . . 86

Chapter 7. Setting Bravo task parameters . . . 89
List of Bravo task parameters . . . 90
Setting Configure Labware (Built In) task parameters . . . 92
Setting location properties for Bravo pipette tasks . . . 94
Setting Aspirate (Bravo) task parameters . . . 96
Setting Dispense (Bravo) task parameters . . . 99
Setting Mix (Bravo) task parameters . . . 102
Setting Move to Location (Bravo) task parameters . . . 105
Setting Pump Reagent (Bravo) task parameters . . . 106
Setting the Reserve Location task parameters . . . 108
Setting Serial Dilution (Bravo) task parameters . . . 109
Setting Set Head Mode (Bravo) task parameters . . . 117
Setting Shake (Bravo) task parameters . . . 118
Setting Sub Process (Bravo) task parameters . . . 119
Setting Tips On (Bravo) task parameters . . . 120
Setting Tips Off (Bravo) task parameters . . . 121
Setting User Message task parameters . . . 123
Setting Vacuum filter task parameters . . . 124
Setting Wait For User (Bravo) task parameters . . . 126

Setting Wash Tips (Bravo) task parameters 127
Setting Device Selection in task parameters 131
Setting Advanced Settings task parameters 132
Using JavaScript in VWorks 134
JavaScripts task object and properties 136

Chapter 8. Setting labware definitions 139
About defining labware with the labware editor 140
Labware editor overview 142
Opening the labware editor 145
Adding a labware entry 147
Deleting a labware entry 149
Renaming a labware entry 150
Copying a labware entry 151
Defining general properties 152
Defining plate properties 153
Defining BenchCel properties 156
Defining stacker properties 158
Defining pipette/well properties 160
Defining Bravo properties 162
Inserting an image 163
Defining labware classes 164

Chapter 9. Setting liquid-handling definitions 169
About the liquid library editor 170
Opening the liquid library editor 172
Creating a liquid class 173
Performing an accuracy calibration 175

Chapter 10. Administrator procedures 179
About user accounts and privileges 180
Adding and deleting a user account 181
Moving or sending a registry file 183

Chapter 11. Maintenance and troubleshooting 185
About error handling 186
Compilation warnings and errors 187
Resolving device initialization errors 188
Importing a log file to Excel 190

Index 193

Introduction

1

This chapter introduces the *VWorks4 User Guide*.

Before you use VWorks to operate your lab automation system, become familiar with the contents of this guide.

This chapter contains the following topics:

❑ "Who should read this guide" on page 2
❑ "What this guide covers" on page 2
❑ "What is new in this version" on page 3
❑ "Accessing Velocity11 user guides" on page 4
❑ "Finding your version numbers" on page 6
❑ "Reporting VWorks problems" on page 7
❑ "Sending a bug report" on page 8

Who should read this guide

Job roles

This user guide is intended to be read by people with the following job roles:

Job role	Responsibilities
Integrator	Someone who writes software and configures hardware controlled by VWorks.
Lab manager, administrator, or technician	Someone who is responsible for: ❑ Developing the applications that are run using VWorks ❑ Solving the more challenging problems that might arise ❑ Developing training materials and standard operating procedures for operators
Operator	Someone who performs the daily production work using VWorks and solves routine problems. Your organization may choose to create its own procedures for operators including the procedures in this guide.

What this guide covers

What is covered

This guide covers a description and detailed procedures for using VWorks4 version 3.

It does not provide procedures for operating devices or for using diagnostics software.

Additional information

For information on specific devices and how to use the diagnostic software for those devices, see the user guide for the individual device.

For information on how to set up third-party devices, see the *Device Driver User Guide*.

Supported software version

This guide documents VWorks4, version 3. For documentation covering earlier versions of VWorks, go to http://www.velocity11.com.

Related topics

For information about...	See...
What is new in this version of the manual	"What is new in this version" on page 3
Accessing Velocity11 user guides	"Accessing Velocity11 user guides" on page 4

What is new in this version

About this topic This topic briefly describes the new features that this user guide covers.

New features

Feature	Description	See
New user interface	VWorks4	"Overview of the VWorks user interface" on page 14
VWorks Instrument	VWorks4 license enabling you to run a Bravo	"Description of VWorks" on page 12
VWorks Benchtop	VWorks4 license enabling you to integrate the PlateLoc and the BenchCel with the Bravo	"Devices you can use with VWorks" on page 13

Related topics

For information about...	See...
What this guide covers	"What this guide covers" on page 2
Accessing Velocity11 user guides	"Accessing Velocity11 user guides" on page 4

Accessing Velocity11 user guides

About this topic

This topic describes the different formats of Velocity11 documentation.

Each Velocity11 user guide is delivered to you as:

- ❑ Online help
- ❑ A PDF file
- ❑ A printed book

The information in each format is the same but each format has different benefits.

Where to find the user guides

Online help

The online help is added to your computer with the software installation.

Velocity11 website

You can download the latest version of any PDF file from our website at www.velocity11.com.

All Velocity11 user documentation can be searched from the website at www.velocity11.com.

Online help

The online help is the best format to use when you are working at the computer and when you want to perform fast or advanced searches for information.

To open the online help:

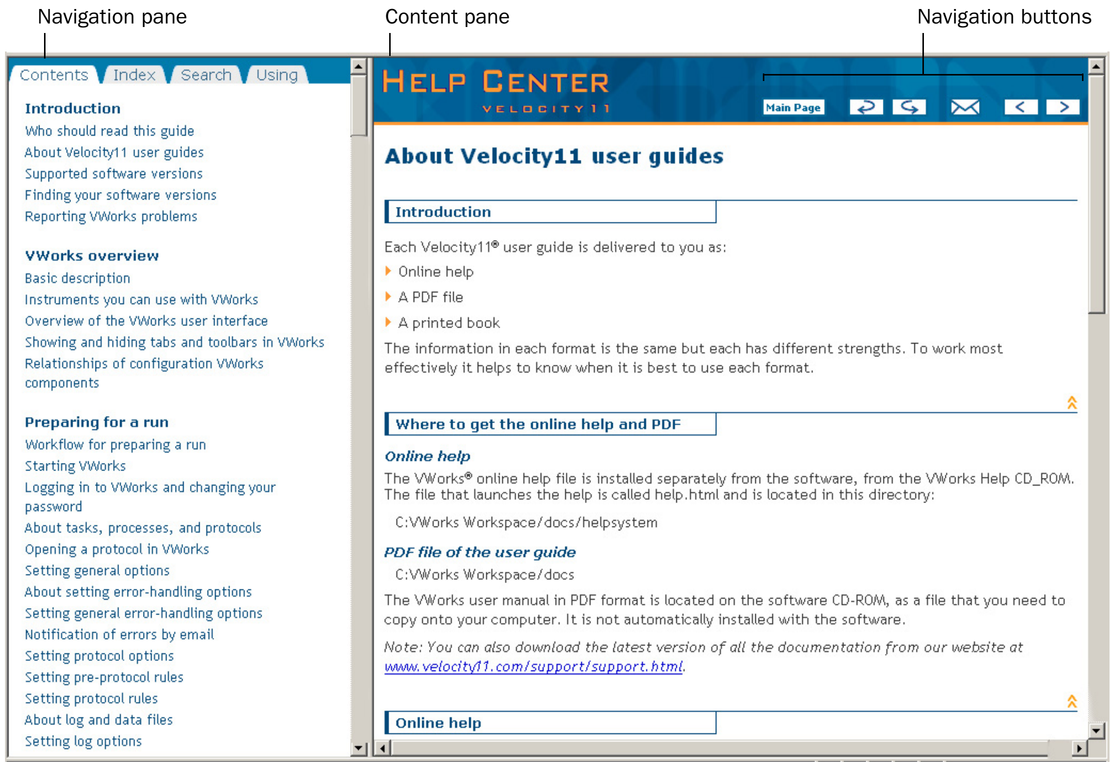
1. If you are currently using VWorks click:
   - ◆ The question mark button to toggle on the help mode or,
   - ◆ Anywhere in the application

     Follow the links to take you to either the relevant part of the VWorks help or the device drivers help.

     Note: You can go directly to the VWorks help by choosing **Help** > **VWorks Help**.
2. If you are not currently using VWorks, open help.html in both:
   - ◆ C:\VWorks Workspace\bin\HelpSystem\vworks_ug for the main help
   - ◆ C:\VWorks Workspace\bin\HelpSystem\devicedriver_ug for the device driver help

Main features

The online help includes a navigation pane, content pane, and navigation buttons.



The navigation pane has four tabs. The Contents, Index, and Search tabs provide different ways to locate information. The Using tab contains information about using the help system. The content pane displays the online help topics.

Navigation buttons in the content pane allow you to navigate through the pages.

PDF user guides

Computer requirements

To open a user guide in PDF format, you need an Acrobat viewer. You can either use the viewer that is built into Adobe Acrobat, or you can download the free Adobe Reader application from http://www.adobe.com/support/downloads/main.html.

Printing and searching

We provide user guides in PDF format mainly for printing additional copies. You can use them for simple searches from the Find button, although these searches are much slower than online help searches:

More information

For more information about using PDF documents, see the Adobe Acrobat PDF help system that can be accessed from your Acrobat viewer.

Related topics

For information about...	See...
What this guide covers	“What this guide covers” on page 2
What is new	“What is new in this version” on page 3
VWorks4 overview	“Description of VWorks” on page 12

Finding your version numbers

About this topic

This topic shows you some ways to find out your version of VWorks, the VWorks device driver plug-in, and VWorks firmware.

From the software

To find the VWorks or support file version number:

1. Start VWorks.
2. Select **Help > About VWorks**.

To find the firmware version number:

1. Start VWorks.
2. Open diagnostics for the device.

   Click **About** and read the version number in the **About VWorks Control** dialog box.

From the files

You can look at the VWorks and device driver version information in the executable files.

To find the VWorks version number:

1. Navigate to C:\Program Files\Velocity11\VWorks.
2. Right-click VWorks4.exe.
3. Select **Properties**.
4. Click the **Version** tab.

To find the device driver version number:

1. Navigate to C:\Program Files\Velocity11\VWorks\plugins.
2. Right-click .dll file for the device.
3. Select **Properties**.
4. Click the **Version** tab.

Related topics

For information about...	See...
What this guide covers	"What this guide covers" on page 2
What is new	"What is new in this version" on page 3
VWorks4 overview	"Description of VWorks" on page 12

Reporting VWorks problems

About this topic

If you find a bug in the software or have a technical or hardware problem that you cannot resolve after reading the chapter on maintenance and troubleshooting, read the information in this topic for how to report problems.

Reporting software problems

If you find a problem in the Velocity11 software, let us know by:

- ❑ Sending a bug report from within VWorks
- ❑ Sending an email to service@velocity11.com or euroservice@velocity11.com

Sending files

When resolving software bugs or other problems, we might ask you to send device and protocols files and the Velocity11 registry file from the Windows registry.

Reporting user guide problems

If you find a problem with this user guide or have suggestions for improvement, please take a minute or two to give us your feedback using the feedback button in the online help. Your comments will be reviewed promptly and used to write the next version of the guide.

You can also send an email directly to documentation@velocity11.com.

Related topics

For information about...	See...
Sending registry files	"Moving or sending a registry file" on page 183
Reporting a bug	"Sending a bug report" on page 8
Setting up email	"Notification of errors by email" on page 32

Sending a bug report

About this topic

This topic describes how to send a bug report to Velocity11 from VWorks.

Requirements

Before you can send a bug report:

- ❑ The system's computer must be connected to a network with internet access.
- ❑ The outgoing email server must be set up on the system's computer by a VWorks or network administrator.

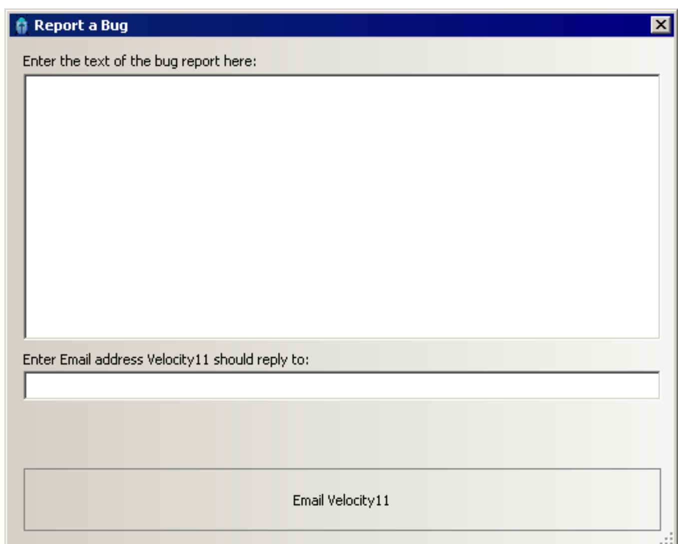
Sending a bug report

A bug report is an email that you create and send from within VWorks. The email is sent directly to Velocity11.

To send a bug report:

1. Select **Help** > **Report a Bug**.

   The **Report a Bug** dialog box opens.



2. Type a description of the error in the text box.

   In your description, provide a summary of the error and, in the case of a software bug, a description of how we can reproduce it.
3. Click **Email Velocity11** and wait until a **Message Sent** message box opens.

Related topics

For information about...	See...
Setting up email	"Notification of errors by email" on page 32

For information about...	See...
Reporting a VWorks problem	“Reporting VWorks problems” on page 7
User accounts and privileges	“About user accounts and privileges” on page 180

VWorks overview

2

This chapter introduces VWorks and its user interface.

This chapter contains the following topics:

- “Description of VWorks” on page 12
- “Devices you can use with VWorks” on page 13
- “Overview of the VWorks user interface” on page 14
- “Relationships of VWorks components” on page 24

Description of VWorks

About this topic

This topic briefly introduces VWorks.

VWorks license types

VWorks is software that manages and controls lab automation systems. VWorks can control very simple systems, such as a single pipettor that is fed microplates (plates) by a human, and it can control complex systems that use a robot to move plates between one or more devices.

Two VWorks license types are available: Instrument and Benchtop. With the Instrument license, you can use VWorks to control a single device, such as the Bravo. With the Benchtop license, you can use VWorks to control multiple devices at once.

Be aware that some of the software interface is different between these two license types and therefore some features and procedures described in this user guide might not be relevant to your license.

About protocols

Users create and run protocols, which are sequences of tasks. Each task performs an activity, such as putting tips on a pipette head or moving a plate to a plate sealer.

When operating VWorks with a Benchtop license, VWorks is a multithreaded application, enabling a single protocol to simultaneously run more than one sequence of tasks.

Event-driven controller

Rather than a dynamic or static scheduler, VWorks uses an event-driven controller to run protocols. This means that protocols are run with no pre-set schedule. An event-driven controller analyzes the protocol as it is run and performs the tasks in a manner that uses the instruments simultaneously and most efficiently. This efficiency reduces the overall time of the run.

Protocols are compiled and simulated

Before a protocol is run, a compiler checks it for logical errors that would otherwise prevent the protocol from completing. Users also have the option of running simulations of the protocol before committing expensive samples.

Real-time manipulation and troubleshooting

VWorks can also manipulate each instrument in the system in real-time by sending individual commands using diagnostics software. This is useful for setting up and troubleshooting.

Comprehensive event logging allows the operator to analyze each run and troubleshoot problems.

21 CFR compliance

VWorks complies with the United States code of regulations Rule 21 CFR Part 11 to ensure the integrity of electronic records. All operations performed are written to an output file. An authorized administrator has the ability to limit users access to the system and records. For example, a user logged in as an operator may have the privilege to run protocols but not to create or edit them, while another type of user, such as a technician, can do both. All events are recorded with a time stamp and cannot be deleted or edited from within VWorks.

Related topics

For information about...	See...
Devices you can use with VWorks	"Devices you can use with VWorks" on page 13
The user interface of VWorks	"Overview of the VWorks user interface" on page 14
Components of VWorks	"Relationships of VWorks components" on page 24

Devices you can use with VWorks

About this topic

This topic lists the devices that you can use with VWorks.

If there is a product not listed here that you would like to add to your lab automation system, please contact Velocity11 Service Center for customization information.

Velocity11 devices

The following Velocity11 devices can be used with VWorks.

Device	Description and comments	Requires this VWorks license
BenchCel	Plate-processing automation platform that stores plates and moves them to and from separate devices	Benchtop
Bravo	Nine-location liquid handler	Instrument or Benchtop
PlateLoc	Thermal plate sealer	Benchtop
Platepad	Additional location for placing labware	Benchtop

Related topics

For information about...	See...
What a device file is	"Overview of the VWorks user interface" on page 14
What a device is	*Device Driver User Guide*
How device files are used	"Relationships of VWorks components" on page 24

Overview of the VWorks user interface

About this topic

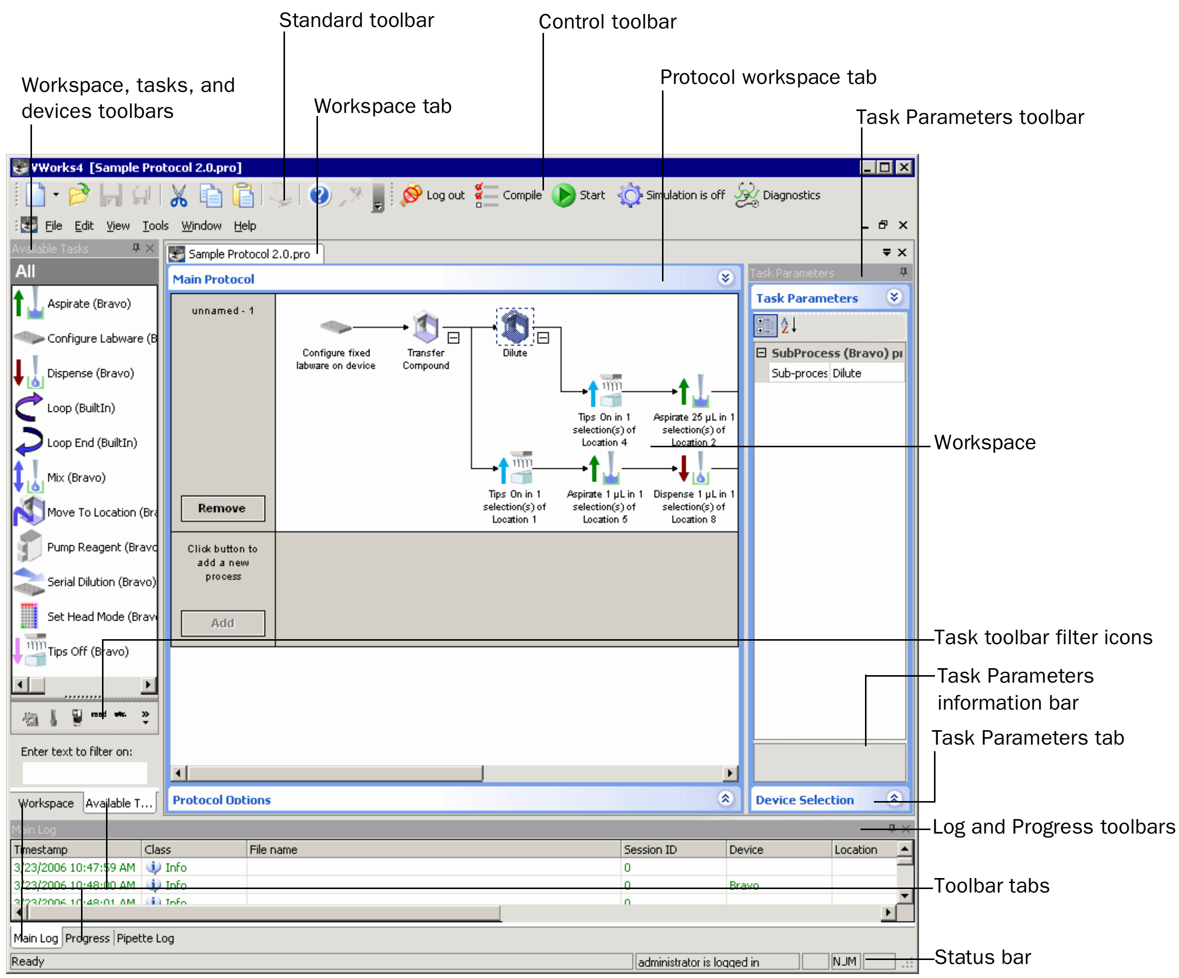
This topic introduces the pages, toolbars, and menus that make up the VWorks user interface (UI).

About the VWorks UI

The VWorks UI is made up of a tabbed workspace, tabbed toolbars, a menu bar, and a status bar. The content of these items can change depending on whether you are viewing a device file or protocol file.

UI terminology

The following diagram identifies the elements of the VWorks UI.



Controlling the UI appearance

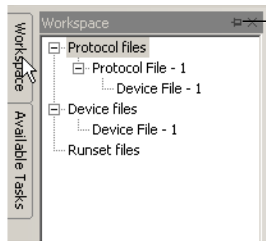
The toolbars can be moved or hidden to suit your preference.

To display or hide a toolbar:

1. Select **View > *item***. If a check mark is displayed next to the item, it is displayed in the user interface. If no check mark is displayed next to the item, it is hidden.

Note: You can also access this menu by right-clicking the protocol workspace.

Alternatively, you can convert the toolbars to tabs by clicking the pushpin. When you rest the pointer on a tab, it opens the toolbar.



To move a toolbar:

1. Drag the title of a toolbar.

   The toolbar undocks from its position and you can move it freely.

2. Re-dock the toolbar:

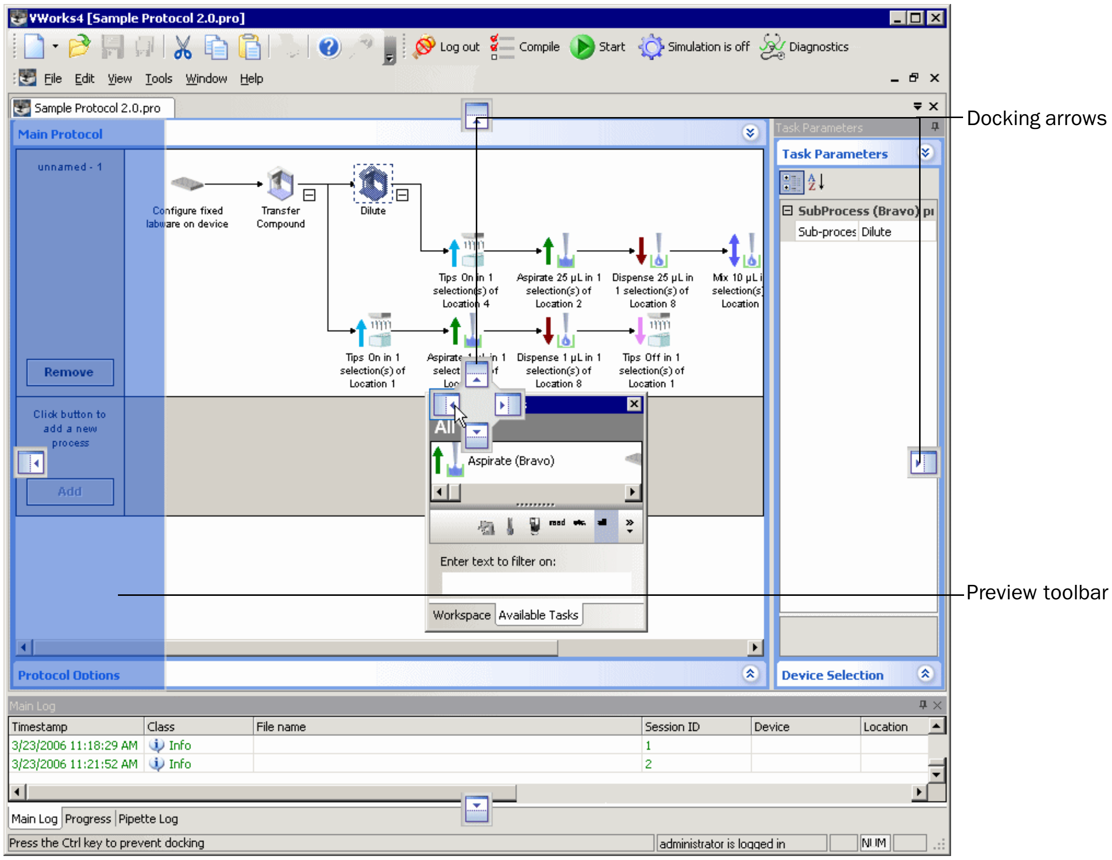
   a. Drag the toolbar title to display the docking arrows.

   b. Select a docking position by dragging the toolbar until the pointer is directly over an arrow.

      A preview of where the toolbar will dock appears.

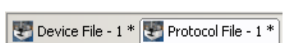
   c. Release the toolbar.

      The toolbar docks to the selected position.



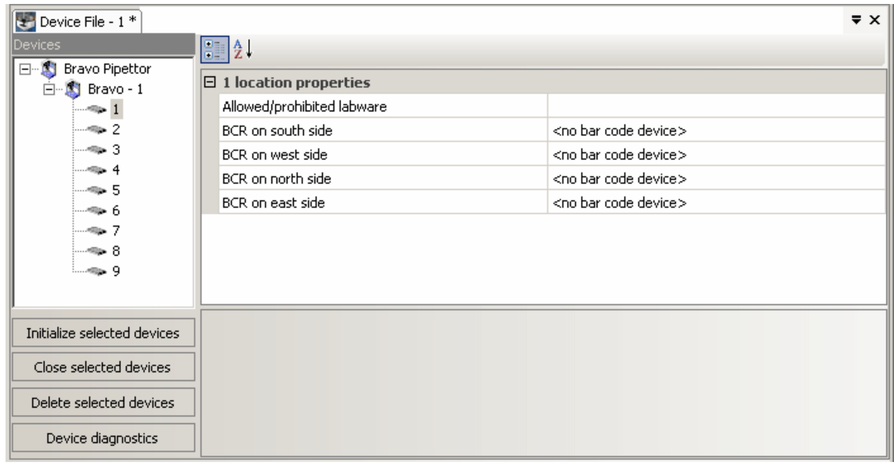
Device and protocol file workspace

The device and protocol file workspaces are accessible by tabs when you open or create a device or protocol file. When multiple protocol files or device files are open, each has its own tab.



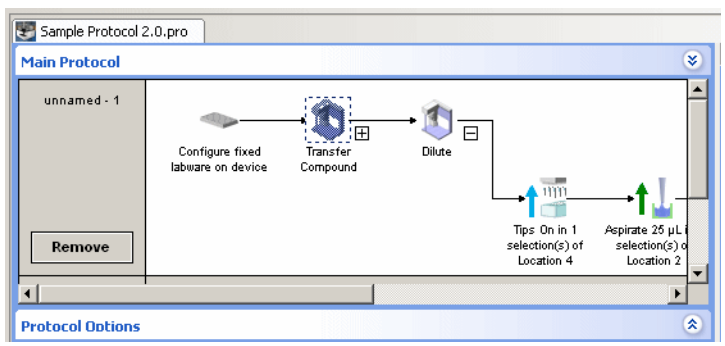
Device File

The Device File tab is accessible when you open or create a device file. The device file displays a list of the devices and their properties that are in that device file. This is where you can edit and save the device properties and access diagnostics for the devices.



Protocol File

The Protocol File tab is accessible when you open or create a protocol file. The protocol file is used to create, save, and then run protocols. Protocols are sequences of tasks that determine how your plates are manipulated.



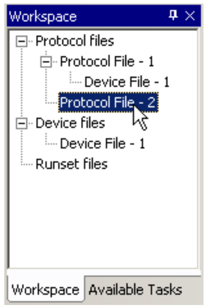
The Protocol File has two to four tabs:

- ❑ *Startup Protocol*. This is an editor used to create the startup protocol. This tab is only available if you are controlling multiple instruments with VWorks.
- ❑ *Main Protocol*. This is an editor used to create the protocol.
- ❑ *Protocol Options*. This specifies additional information associated with the protocol. For example, this is where you set the device file for the protocol.
- ❑ *Cleanup Protocol*. This is an editor used to create the cleanup protocol. This tab is only available if you are controlling multiple instruments with VWorks.

Toolbars

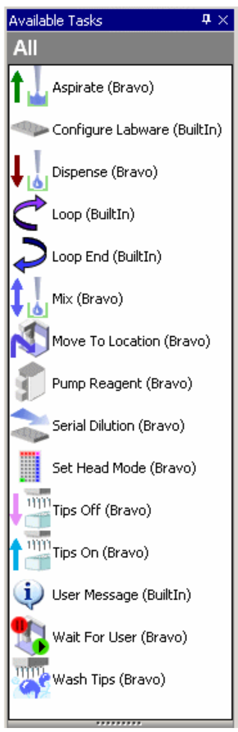
Workspace toolbar

The Workspace toolbar is accessible when viewing either device or protocol files. The Workspace toolbar displays all of the device and protocol files that are currently open and enables you to quickly move from one file workspace to another.



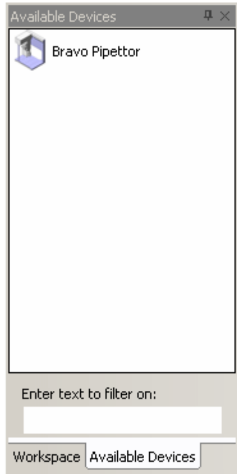
Available Tasks toolbar

The Available Tasks toolbar is only accessible when you are viewing a protocol file. This toolbar displays all of the tasks available for making protocols. The task icon display can be filtered by clicking one of the icons at the bottom of the toolbar or by entering a string of text.



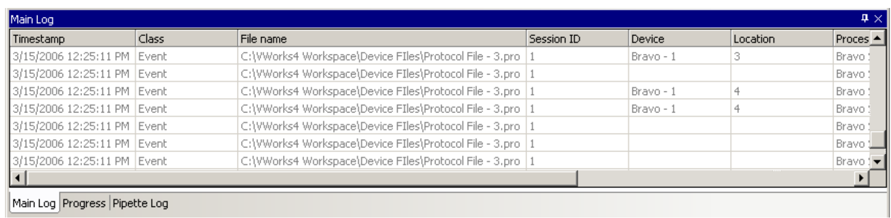
Available Devices toolbar

The Available Devices toolbar is only accessible when you are viewing a device file. This toolbar displays all of the devices that VWorks can communicate with, if connected to the controlling computer. When you double-click a device in this list, a new device file containing the device is automatically created.



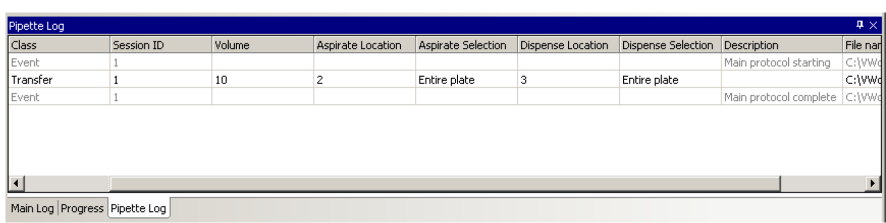
Main Log toolbar

The Main Log toolbar displays all the events that occur when VWorks is running. These events are automatically written to a text file in accordance to Rule 21 CFR Part11 and can be accessed at any time. Click any column to sort log events according to that category.



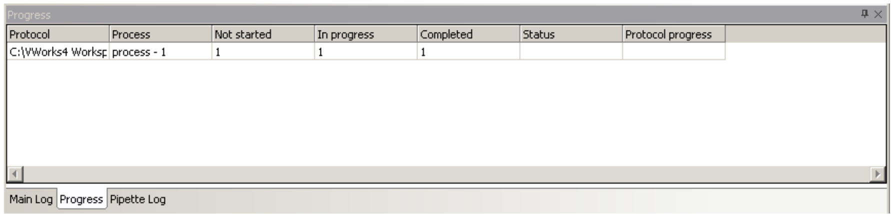
Pipette Log toolbar

The Pipette Log toolbar displays all the details specific to each pipette transfer task that is performed when VWorks is running. These details are written to a text file in accordance to Rule 21 CFR Part 11 and can be accessed at any time. Click any column to sort pipette log events according to that category.



Progress toolbar

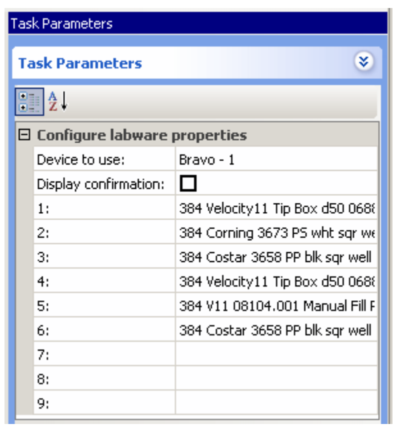
The Progress toolbar contains information about the status and progress of the protocol.



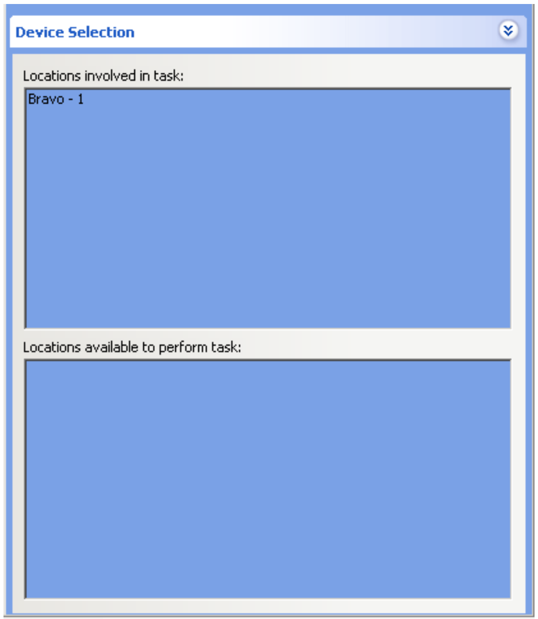
Task Parameters toolbar

The Tasks Parameters toolbar is only accessible when viewing protocols. The Task Parameters toolbar has three tabs:

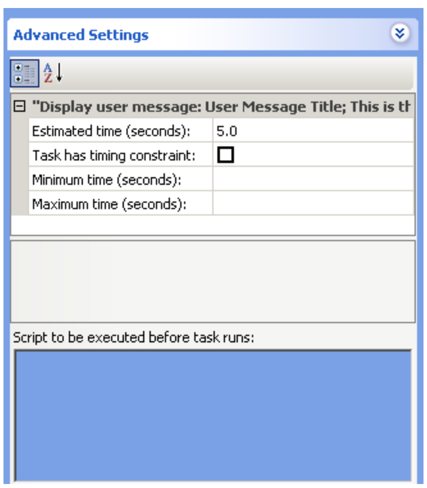
- ❑ *Task Parameters*. Displays the settings for the task that is currently selected in the protocol editor. Setting the parameters for a specific task is performed here.



- ❑ *Device Selection*. Displays all of the devices that are scheduled to perform the selected task and the devices that are available to perform the selected task. This page is typically used when there are multiple devices connected and there is a choice of which device to use for a task.



- *Advanced Settings*. Enables you to specify a time estimate for the task for use by the protocol simulator. Also, in future releases, you will use Advanced Settings to enter a script to be executed for that task. For example, it can instruct the pipettor to move lower during multiple aspirations from a set of wells.



Standard toolbar

The Standard toolbar contains buttons for common commands, such as creating new protocol files, copying, cutting, and pasting. You can get information about each button by resting your pointer over it.

Control toolbar

The buttons on the Control toolbar have a variety of uses in VWorks and are documented as needed in this guide.

Menus

Each of the menus is briefly described here.

File

Use the commands on the File menu to:

- Create, open, close, and save files
- Close VWorks

Edit

Use this menu for cutting, copying, pasting, and deleting tasks in a protocol.

View

Use the commands on this menu to toggle the display of the toolbars and status bar.

Tools

Use the commands on this menu to perform tasks specific to VWorks. These commands are documented as needed in this guide.

Window

Use this menu to view another file that is open but not currently displayed.

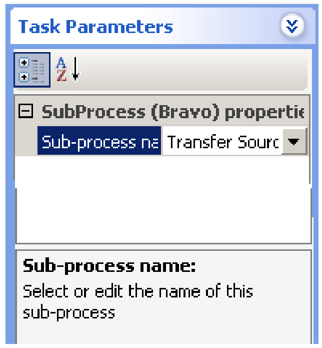
Help

Use the commands on this menu to:

- Open the online help for VWorks
- Show the version of VWorks and the support .dll files
- Report a bug to Velocity11

Task Parameter information bar

The Task Parameters information bar is located on the bottom of the Task Parameters toolbar. It displays information about the property currently selected in the Task Parameters toolbar.



Status bar

The Status bar is located on the bottom of the running VWorks workspace. It displays the current state of VWorks, the current user, and tooltips.

Related topics

For information about...	See...
Starting VWorks	“Starting VWorks” on page 29
Components of VWorks	“Relationships of VWorks components” on page 24
General description of VWorks	“Description of VWorks” on page 12

Relationships of VWorks components

About this topic

This topic introduces important components used by VWorks: the protocol file, the device file, the profile, the teachpoint file, the labware database, the liquid library database, and the user database.

The topic describes what these components are and how they work together to operate your lab automation system.

What you should know

It is important to understand the way each of the main components in VWorks relate. Loading some components automatically loads others. For example, loading a protocol file loads a device file.

Definitions

Refer to the table below for information about VWorks configuration components:

Component	Definition	See
Protocol file	A file that contains instructions for performing a run.	"About tasks, processes, and protocols" on page 68
Device file	A file that contains configuration information for configured devices.	❑ *Device Driver User Guide* ❑ *Bravo User Guide*
Profile	A collection of settings, stored in the Windows registry, that manages how you connect to devices.	The relevant hardware user guide
Teachpoint file	A file that saves your teachpoint settings.	The documentation provided with the robot you are using
Labware database	Labware definitions and classes stored in the Windows registry.	"Setting labware definitions" on page 139
Liquid library database	Pipetting settings, setup for different liquid types, stored in the Windows registry.	"Setting liquid-handling definitions" on page 169
User database	List of user accounts, privileges, and passwords stored in the Windows registry.	"Administrator procedures" on page 179

Configuration component relationships

Refer to the table below to understand the relationships of these VWorks configuration components:

Component	Extension	Opening this file loads...
Protocol	.pro	❑ VWorks (if it is not already running) ❑ Device file ❑ Profile for each device ❑ Teachpoint file
Device File	.dev	❑ Profile for each device ❑ Teachpoint file
Profile	None	Teachpoint file (you are prompted)
Teachpoint File	.txt	Teachpoint definitions

Related topics

For information about...	See...
Exporting and importing Windows registry files	"Description of VWorks" on page 12
VWorks user interface	"Overview of the VWorks user interface" on page 14
Devices you can use with VWorks	"Devices you can use with VWorks" on page 13

Preparing for a run

3

The procedures in this chapter can be performed by someone with operator privileges.

All procedures in this chapter assume that VWorks has been installed and set up, and that protocols have been created.

This chapter contains the following topics:

- Workflow for preparing a run
- Starting VWorks
- Logging in to VWorks and changing your password
- Opening a protocol in VWorks
- Notification of errors by email
- Setting log file directories
- Understanding the protocol

Workflow for preparing a run

About this topic

This topic lists the procedures that you need to follow before performing a run in VWorks.

Workflow

The general workflow for preparing for a run is given in the following table:

Step	Topic
1	“Starting VWorks” on page 29
2	“Logging in to VWorks and changing your password” on page 30
3	“Opening a protocol in VWorks” on page 31
4	“Notification of errors by email” on page 32
5	“Setting log file directories” on page 34
6	“Understanding the protocol” on page 35

Related topics

For information about...	See...
Performing a run in VWorks	“Performing a run” on page 37

Starting VWorks

About this topic

This topic describes how to start VWorks.

Before you start

Before you start, turn on the devices that you are using with VWorks.

See the relevant device user guide for instructions.

Starting VWorks

To start VWorks:

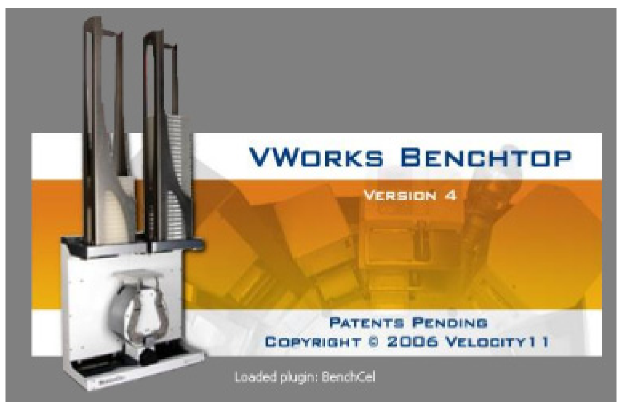
1. Make sure that everyone is clear of the lab automation system and that there are no objects that could obstruct any moving parts.

   !! INJURY HAZARD !! When VWorks starts, device parts might unexpectedly move to their home positions.

2. Double-click the shortcut to VWorks on the Windows desktop.

   Note: If the shortcut has been deleted, open the folder C:\Program Files\Velocity11\VWorks and create a new shortcut from the executable file VWorks*version*.exe.

   The VWorks splash screen opens.



Related topics

For information about...	See...
What to do next	"Logging in to VWorks and changing your password" on page 30
The workflow that this procedure belongs to	"Workflow for preparing a run" on page 28

Logging in to VWorks and changing your password

About this topic

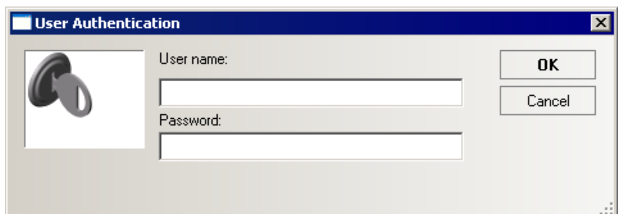
This topic describes how to log in to VWorks and, if necessary, change your password.

Logging in

To log in to VWorks:

1. Click **Log in**.

2. Enter your account **User Name**.



3. Enter your password and click **OK**.

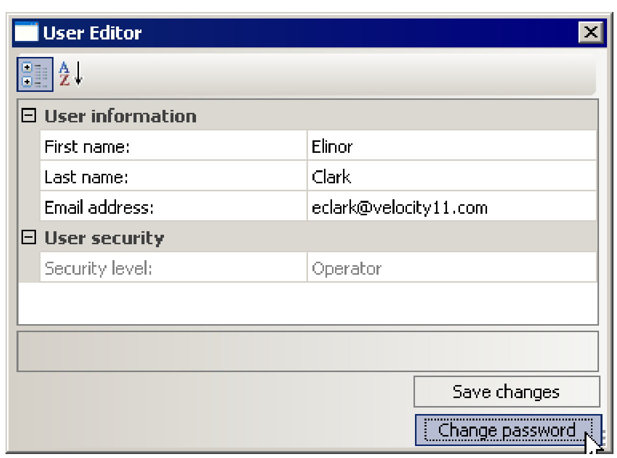
   Note: If this is the first time anyone has logged in to this installation of VWorks, there is one administrator account called administrator and its password is administrator.

Changing your password

If you are an operator or a technician, you can change the password for your user account by following this procedure.

To change your user account password:

1. Select **Tools** > **User Management**.
2. Click **Change Password**.



3. In the **Change Password** dialog box, enter your old and new passwords and click **OK**.
4. Close the **User Editor** dialog box.

Related topics

For information about...	See...
What to do next	“Opening a protocol in VWorks” on page 31
The workflow that this procedure belongs to	“Workflow for preparing a run” on page 28
Managing users	“Adding and deleting a user account” on page 181

Opening a protocol in VWorks

About this topic

Use the procedure in this topic to open a protocol that has already been created.

Procedure

To open a protocol:

1. Select **File** > **Open**.

   Make sure you set the file type to Protocol (.pro) when browsing for the file.

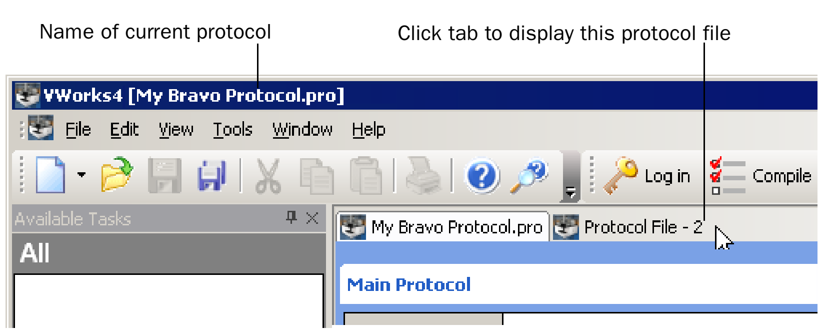
You can also open a protocol by navigating to the .pro file in Windows and double-clicking it. This launches VWorks and opens the protocol.

Multiple protocols

In VWorks, you can have more than one protocol open at a time. The name of the current protocol appears in the title bar.

To move between multiple protocols:

1. Click the protocol tab you want to view.



Related topics

For information about...	See...
How to create a protocol	"Creating a protocol" on page 67
Associate a device file with a protocol	"Setting protocol options" on page 75
The workflow that this procedure belongs to	"Workflow for preparing a run" on page 28
VWorks components	"Relationships of VWorks components" on page 24

Notification of errors by email

About this topic

This topic describes how to add an email address to VWorks so you can be notified by email or pager when there is a run error.

Email setup in VWorks enables you to do the following tasks:

- ❑ Automatically be notified by email, pager, or text message on your mobile phone when errors occur during a protocol run
- ❑ Send a bug report to Velocity11

Requirements for email setup

Before you can send an email from VWorks:

- ❑ The VWorks computer must be connected to a network with internet access
- ❑ The outgoing email server must be set up on the system's computer

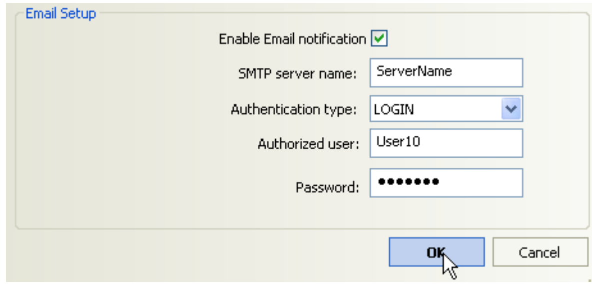
Setting up email

To set up email notification:

1. Select **Tools > Options**.
2. In the **Email Setup** section, select **Enable Email notification** check box.
3. In the **Email Setup** section, enter the name of your **SMTP server name** (outgoing email server).

   Note: To be notified on your mobile phone with a text message, enter the appropriate email address for your mobile phone number. The typical format is: 10_digit_number@mobile_carrier_domain.com. Check with your mobile phone service provider. Any charges you might incur, and the way messages are delivered and displayed depends on your wireless device and service plan.
4. If the server requires a user name and password:
   a. Select the **Authentication type** from the list.

b. Enter your **Authorized user** name and **Password** for the selected authentication type.



This information only needs to be set up once, provided the email account remains active. Emails sent from VWorks is authenticated using this account.

Related topics

For information about...	See...
What to do next	“Setting log file directories” on page 34
The workflow that this procedure belongs to	“Workflow for preparing a run” on page 28

Setting log file directories

About this topic

This topic describes how to change the log file location. There are two log types:

- *Pipette log.* The pipette log records all pipetting transfer tasks.
- *Main log*. The main log records all of the actions that occur in VWorks.

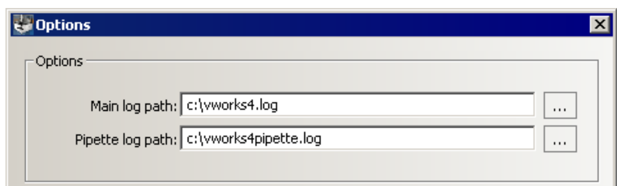
You cannot edit or delete log entries within VWorks, but you can decide where they are stored on your hard drive.

Changing the log files location

To change the location for storing your log files:

1. Select **Tools > Options**.
2. To change the default location, click the browse button, navigate to the desired directory and click **Save**.

   The path to the folder where the log files are saved is displayed.



Related topics

For information about...	See...
What to do next	“Understanding the protocol” on page 35
The workflow that this procedure belongs to	“Workflow for preparing a run” on page 28
Importing a log file to Microsoft Excel	“Importing a log file to Excel” on page 190

Understanding the protocol

About this topic

If you are not already familiar with the protocol you intend to run, you need to take some time to understand it.

What you should know

At a minimum, become familiar with the following:

- ❑ Which devices you need to prepare
- ❑ Where you need to position the plates before the run and where they are moved to during the run
- ❑ Whether User Message tasks prompt you to perform certain actions after you start the run or whether you need to perform the actions on your own initiative before you start the run
- ❑ Whether you need to replace fluids and empty waste during the run
- ❑ Whether you need to remove and add plates during the run
- ❑ Which liquids you need to prepare, where they should be placed, and in what kinds of reservoir

Related topics

For information about...	See...
Creating a protocol	"Creating a protocol" on page 67
The workflow this topic belongs to	"Workflow for preparing a run" on page 28
Starting a run	"Performing a run" on page 37

Performing a run

4

This chapter contains the following topics:

- "Workflow for performing a run" on page 38
- "Starting a run from VWorks" on page 39
- "Pausing or stopping a run" on page 41
- "Monitoring a run" on page 43
- "Working with the Log toolbars" on page 44
- "Closing down after a run" on page 46

Workflow for performing a run

About this topic

This topic gives the typical procedures you need to follow to perform a run.

Workflow

Step	Topic
1	"Starting a run from VWorks" on page 39
2	"Pausing or stopping a run" on page 41
3	"Monitoring a run" on page 43
4	"Working with the Log toolbars" on page 44
5	"Closing down after a run" on page 46

Related topics

For information about...	See...
What to check before a run	❑ "Understanding the protocol" on page 35 ❑ Appropriate device user guide
Opening a protocol	"Opening a protocol in VWorks" on page 31
Changing protocol options	"Setting protocol options" on page 75

Starting a run from VWorks

About this topic

This topic describes how to start a run using the Start button in VWorks. Start a run after performing pre-run checks and opening a protocol.

Procedure

To start a run:

1. Make sure all devices used in the protocol are in their home positions.

   Refer to the device user guides for more information about homing.

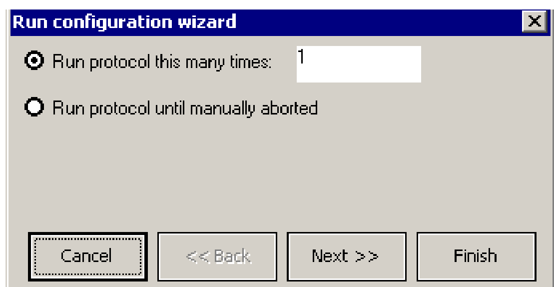
2. Make sure the areas around the devices are clear of labware (except for the labware used in the protocol).

3. In VWorks, click **Start**.

   If this is the first run with this device file, VWorks confirms communication with all devices and instructs the devices to home.

   If there are no protocol compilation errors, the **Run configuration wizard** opens.

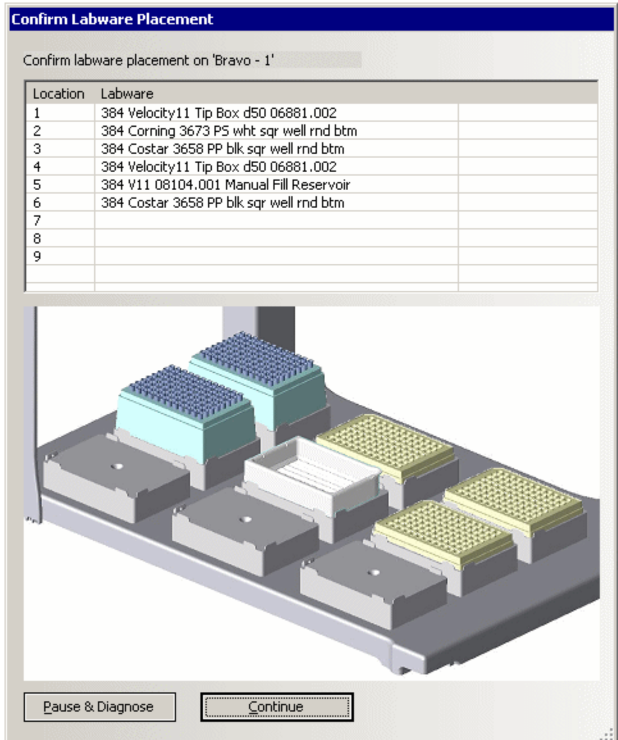
4. Type the number of times that you want to execute the protocol during the run, or select **Run protocol until manually aborted**.

5. Click **Next** to set up bar code tracking, or click **Finish** to close the wizard.

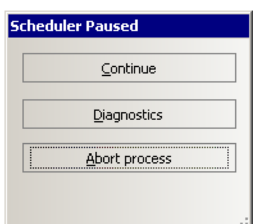
   If you click **Finish**, the run starts unless **Display confirmation** is selected in the **Configure Labware** task parameters.

   If **Display confirmation** is selected, the **Confirm Labware Placement** dialog box opens.



6. If the display shows the correct labware, click **Continue** to start the run, otherwise click **Pause & Diagnose**.

   After clicking **Pause & Diagnose**, you have the option of continuing with the protocol, opening diagnostics, or aborting the run.



If **User Message** tasks are included in the protocol, you are prompted to respond to them as they execute.

If there are no **User Message** tasks to remind you to empty liquid waste containers and refill liquid source containers, set your own reminders using lab timers.

What happens after a run starts

After you start the run:

- ❑ The Start button becomes unavailable and the Pause button becomes available.
- ❑ Log messages on the Log toolbar indicate the start of the run. Also, log files record events as they are performed on all plates in the run.

- ❑ If you selected the option for the robot to check all plate positions for plates, this operation will now be performed. If an error is found, correct the problem and click Retry.
- ❑ The first instructions of the protocol are executed.
- ❑ If User Message tasks are included in the protocol, you are prompted to respond to them.

Related topics

For information about...	See...
Setting up bar code tracking	"Tracking plates using bar codes" on page 86
Workflow this procedure belongs to	"Workflow for performing a run" on page 38
Pausing a run	"Pausing or stopping a run" on page 41
Monitoring a run	"Monitoring a run" on page 43

Pausing or stopping a run

About this topic

This topic describes when and how to stop or pause a protocol that is running.

Use this procedure to pause and continue a run. For example, pause a run when you want to:

- ❑ Add or remove labware
- ❑ Clean up a spill
- ❑ Add buffer to a reservoir
- ❑ Diagnose a problem
- ❑ Perform an operation that is not part of the protocol

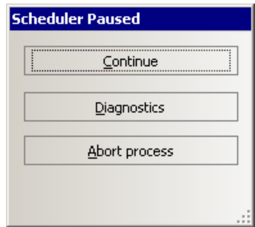
To abort a run in an emergency situation, for example to prevent a robot crash, use the robot disable button. See the appropriate device user guide for the procedure.

Procedure

To pause or stop a run using VWorks:

1. In VWorks, click **Pause**.

   The **Scheduler Paused** dialog box opens and the currently scheduled task continues to completion. This might take a minute or more. After that, no more tasks are performed.



2. You now have three choices:

If you want to...	Then...
Continue with the run	Click **Continue**.
Troubleshoot a problem or perform a manual operation	Click **Diagnostics**, and select the module that caused the error. This opens the diagnostics software for that module, allowing you to troubleshoot the problem.
Abort the protocol	Click **Abort Process**.

!! IMPORTANT !! Before continuing with a run, make sure the system is in a valid state for the protocol.

Make sure you have not made changes that will cause an error, such as moving a plate to a location that should not have a plate.

Related topics

For information about...	See...
Workflow this topic belongs to	“Workflow for performing a run” on page 38
Monitoring a run	“Monitoring a run” on page 43
Using the log	“Working with the Log toolbars” on page 44
Using the robot disable button	User guides for your devices

Monitoring a run

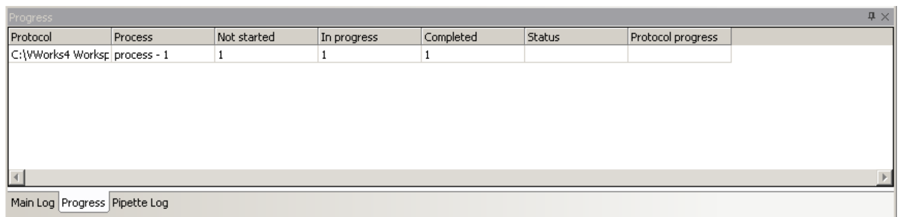
About this topic

After starting a run, the operation of the system should be monitored. This topic describes the general ways to monitor a run and points to other topics that describe how to monitor more specific aspects of a run.

Monitoring overall progress

You can monitor overall progress of the run in the Progress page of VWorks. (Click the **Progress** tab at the bottom of the Log toolbar.)

If you do not see the **Progress** tab, select **View > Progress**.

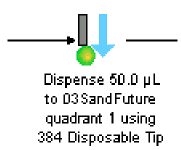


What to monitor

Exactly what you do to monitor a run depends on your lab automation system and the protocol that you are using. For example, you might need to:

- ❑ Compare the motions of the robot to the protocol tasks being completed.

  You can identify the task that is currently being performed from the position of the green ball in the process panes. Because VWorks can schedule more than one task at a time, there might be more than one green ball displayed.

- ❑ Add and remove labware.
- ❑ Empty liquid waste containers.
- ❑ Fill liquid reservoirs.
- ❑ Replace supplies, such as plate seal.

!! IMPORTANT !! No errors are reported when a liquid waste container becomes full or a liquid reservoir becomes empty. (Exceptions to this are reservoirs on a Weigh Station.)

To guard against the problem of a full waste container or empty reservoir container, the protocol writer can incorporate User Message tasks into the protocol to remind the operator at the appropriate steps in the protocol. Alternatively, operators can set timer alarms to remind them to fill reservoirs and empty the waste container at the appropriate time.

Related topics

For information about...	See...
Workflow this topic belongs to	"Workflow for performing a run" on page 38
Pausing a run	"Pausing or stopping a run" on page 41
Using the log toolbar	"Working with the Log toolbars" on page 44
Options for system monitoring	User guides for your devices
Troubleshoot an error	Maintenance and troubleshooting chapter of the user guide for your device.

Working with the Log toolbars

About this topic

A history of events that occur when using the VWorks is recorded in a log file. These events are displayed in the Log toolbar.

Log files are text files that cannot be deleted or edited within VWorks and are typically used for troubleshooting purposes.

There are two types of log files:

- ❑ *Pipette log.* The pipette log records the details of all the liquid transfers.
- ❑ *Main log*. The main log records all of the actions that occur in VWorks.

This topic gives an overview of what the Log toolbars do and how to use their features.

Viewing the log toolbars

To view the log toolbars:

1. Select one of the following:
   - ◆ **View** > **Main Log**.
   - ◆ **View** > **Pipette Log**.

   The toolbar opens at the bottom of the screen.
2. Refer to the table below to view different kinds of log data.

Log toolbar information

The following information is displayed in the Main Log toolbar.

Log column	Displays...
Timestamp	Time and date of the event or error.

Log column	Displays...
Class	All events and error messages.
Device	Device at which the event or error occurred.
Location	Deck location where the event or error occurred.
Process	Name of the protocol process and sub-process that is running.
Task	Task at which the event or error occurred.
Description	Description of the action that is being recorded.
File name	Name of the file. When a file is first opened, the location of the file on the hard drive is displayed.

Message color coding

The log color-coding scheme is listed in the following table:

Color	Meaning
Grey	Standard events with a date stamp or user-added notes
Blue	Liquid transfer events
Orange	Warnings
Red	Error
Green	General information

Related topics

For information about...	See...
Setting the log file directories	“Setting log file directories” on page 34
Opening a log file in Microsoft Excel	“Importing a log file to Excel” on page 190

Closing down after a run

About this topic

This topic outlines some cleanup steps that you should perform after a run ends.

Procedure

Perform this procedure immediately after a run ends.

To close down after a run:

1. If you have administrator or technician privileges and you have modified the protocol, you can save the protocol.
2. Log out of VWorks.

   Click **Log Out**.
3. Optionally, shut down the other devices in the lab automation system.
4. Optionally, exit from VWorks.

   Select **File > Exit**.

Related topics

For information about...	See...
Cleaning up after a run	The user guide for the device or system.
Saving protocols	“Compiling protocols” on page 83
Checking log files	❑ “Setting log file directories” on page 34 ❑ “Understanding the protocol” on page 35

Installing VWorks and networking the controlling computer

5

This chapter is intended for people with administrator or technician privileges.

This chapter contains the following topics:

- "Workflow for installing VWorks and networking the computer" on page 48
- "Computer and networking requirements" on page 49
- "Installing VWorks software" on page 50
- "Choosing between serial and Ethernet connections" on page 51
- "About Ethernet networking" on page 52
- "Workflows for Ethernet networking" on page 55
- "Setting the network card's IP address" on page 56
- "Connecting Ethernet cables" on page 59
- "Checking the network card's IP address" on page 60
- "Installing and starting DHCP server software" on page 61
- "Turning off Ethernet firewall software" on page 63
- "Uninstalling VWorks" on page 65

Workflow for installing VWorks and networking the computer

About this topic

Typically, VWorks will already be installed on your controlling computer by Velocity11. However, there might be times, such as when the software or a device undergoes an upgrade, you will need to install or re-install VWorks. Also, changes in your company's networking structure or device configuration might require you to re-network the computer running VWorks.

This topic outlines the procedures you need to follow to install VWorks and network the controlling computer to your lab automation system and local area network (LAN).

Workflow

The general workflow for installing VWorks:

Step	Topic
1	Making sure your computer meets the "Computer and networking requirements" on page 49.
2	"Installing VWorks software" on page 50.
3	"Choosing between serial and Ethernet connections" on page 51.
4	*Ethernet only.* Following the "Workflows for Ethernet networking" on page 55.

Related topics

For information about...	See...
Uninstalling VWorks	"Uninstalling VWorks" on page 65
Setting up a device	❑ *Device Driver User Guide* ❑ User guide for the device

Computer and networking requirements

About this topic

Before installing VWorks, make sure your computer meets the requirements described in this topic.

Connecting computer to device

The controlling computer can be connected to devices by Ethernet or serial cables. (The controlling computer is running VWorks and controls one or more devices.)

If you connect via Ethernet and the controlling computer is already connected to a company or other LAN, you need a second, separate network card for device-to-computer communication. This second network card allows the device to operate on an isolated network.

Note: If you purchased a desktop computer from Velocity11, the computer has two network cards.

Minimum system requirements

PC system

Any PC capable of running Windows 2000 or Windows XP

Communication interface

- ❑ Dedicated 10BaseT Ethernet port (two network cards if connecting to your company LAN)
- ❑ RS-232 serial port

Monitor

- ❑ Recommended: 1280 x 1024 resolution
- ❑ Minimum requirement: 1024 x 768

Related topics

For information about...	See...
Deciding on serial or Ethernet networking	“Choosing between serial and Ethernet connections” on page 51
The workflow that this procedure belongs to	“Workflow for installing VWorks and networking the computer” on page 48
The next step	“Installing VWorks software” on page 50

Installing VWorks software

About this topic

This topic describes how to start the VWorks installer.

Two procedures are given. If you are installing VWorks:

- ❑ For the first time on the system, use Procedure 1.
- ❑ On a system where a version of VWorks is already installed, use Procedure 2.

Procedure 1

To install VWorks for the first time:

1. Insert the VWorks CD-ROM into your CD-ROM drive.
2. Double-click the setup.exe file.
3. Follow the instructions in the wizard to complete your installation.

Procedure 2

If you are reinstalling or installing a newer version of VWorks, use this procedure.

To upgrade an existing version of VWorks:

1. Exit VWorks.
2. Insert the VWorks CD-ROM into your CD-ROM drive.
3. Double-click the setup.exe file.

   The installer prompts you to remove your old VWorks application. If this does not happen, you need to uninstall VWorks before continuing the new installation. See "Uninstalling VWorks" on page 65. You do not need to delete the Velocity11 registry files.
4. Follow the instructions in the wizard to complete your installation.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflow for installing VWorks and networking the computer" on page 48
Uninstalling VWorks	"Uninstalling VWorks" on page 65

Choosing between serial and Ethernet connections

About this topic

Some devices can be connected to the controlling computer by both serial and Ethernet connections. In most cases, whether you use a serial or Ethernet cable to connect your device depends on the device.

If you have a choice of connection type for your device, reading this topic can help you decide which to use.

Connection options

You can connect one or more devices to the controlling computer using either:

- A serial network
- An Ethernet network

How to choose

Why serial might be better

Connecting the computer to devices using a serial cable is simpler than connecting with an Ethernet cable. With serial, there is no network card to install, no IP address to set up, and no software to install.

Serial connections have another advantage. If your controlling computer has an Ethernet network card, you do not have to use it for the lab automation network so you can use it to connect to your building's general Ethernet network. If you want to set up an Ethernet network, you will have to install a second network card.

Why Ethernet might be better

Ethernet is a faster, more reliable, and more flexible communication technology than serial. Depending on the specifications of your computer, you might experience communication problems with a serial network, in which case you will have to use an Ethernet network.

Another advantage of Ethernet networking is that you can run many devices from a single network port on your computer. Serial connections require one port for every device. If you are planning to network multiple devices, your computer might not have enough serial ports so you will have to add more.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflow for installing VWorks and networking the computer" on page 48
The workflow for Ethernet networking	"Workflows for Ethernet networking" on page 55

About Ethernet networking

About this topic

This topic introduces some important concepts about Ethernet networking.

Network requirements

Network cards

Every device and computer requires a network card to communicate with each other. All Velocity11 devices have a network card built in.

Network connections

Ethernet cables are used to make the physical connections between the device, the controlling computer, and local area network (if you have one).

You can use Ethernet cables to connect the VWorks and the controlling computer in a single-device setup, or you can use the Ethernet cables to connect the VWorks computer to a dedicated lab automation local area network.

Most computers come with one Ethernet network card, and, if you want to use this card to communicate with your lab's general network (for email, internet access, and so on), you need to install an additional card to communicate with your device.

If you plan to connect multiple devices, you need an Ethernet switch to connect the single Ethernet cable attached to your computer and multiple Ethernet cables to connect to each of the devices.

About IP addresses

IP address function

The computer and each device on the network must be assigned a different number, called an IP address. IP addresses allow computers and devices to be uniquely identified on a network so that information can be specifically routed to them.

Ways to assign an IP address

There are two ways to assign an IP address:

- ❑ Statically, by assigning it manually so that it never changes.
- ❑ Dynamically, by automatically assigning a new address every time the computer or device is turned on.

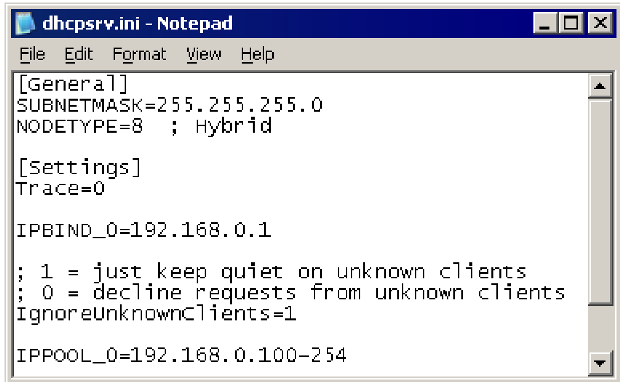
  Dynamic assignment uses DHCP (Dynamic Host Configuration Protocol) software.

The computer's IP address

It is important to consider separately the IP address of the computer and the IP addresses of the devices on the network.

The network card of the computer that is controlling the devices on a lab automation network always has a static IP address. This is even the case when the devices are dynamically assigned addresses because of the way that DHCP servers work. DHCP servers are usually configured to start when Windows starts.

When the server starts, it reads an initialization file that points it to the network card through which the computer will serve addresses. A screenshot of the initialization file is shown below.



Because the IP address and subnet mask in the file is static, the value of the network card's IP address and subnet mask must always match the values in this file. By convention, this address is 192.168.0.1.

!! INJURY HAZARD !! It is possible to run a device on a general network in which both the device and the controlling computer have dynamic IP addresses but Velocity11 cautions against this for safety reasons. Remote computer operators might accidentally initiate an operation that causes the device to move unexpectedly when lab personnel are nearby.

If the computer is also part of a general network, the second card should have a dynamic IP address because the computer does not control the network and the computer is easier to administer this way.

Devices' IP addresses

Devices can have IP addresses that are either static or dynamic.

Choosing an addressing method

Whether to assign the IP address of the device statically or dynamically, depends on the network's architecture.

With a stand-alone connection

If you are connecting a single, stand-alone device that will be set up this way long-term, set the IP address of the device statically (or use a serial connection). This saves you from having to add a second network card to your computer and install DHCP server software.

When integrating into an existing LAN with DHCP

If you are connecting a device to an existing local area network (LAN) that is already running a DHCP server, it is strongly recommended that you dynamically assign the IP address of the device, and leave the statically set IP address of the computer's network card as it is.

When setting up a new LAN connection

If you are setting up a new LAN, you must decide whether to use static or dynamic IP addressing for the device.

In general, if you have more than a few devices on your LAN, it is more convenient to use automatic addressing because you can remove and add devices without having to set up their IP addresses statically each time. If you only have a few devices, it might be more convenient to set them statically and avoid having to set up the DHCP software.

DHCP Server Kit

To make it easier for you to set up an Ethernet network, Velocity11 sells a DHCP Server Kit that includes:

- ❑ A networking card
- ❑ An Ethernet cable
- ❑ An Ethernet switch
- ❑ DHCP server software

The server supplied in the DHCP Server Kit works with computers running a Windows XP or Windows 2000 operating system.

Without the kit, you need to obtain the necessary parts yourself.

Related topics

For information about...	See...
Deciding on serial or Ethernet networking	“Choosing between serial and Ethernet connections” on page 51
Physically connecting the controlling computer to an instrument	“Connecting Ethernet cables” on page 59
Setting a static IP address for an Ethernet connection to a stand-alone instrument	“Setting the network card’s IP address” on page 56
The workflow that this procedure belongs to	“Workflow for installing VWorks and networking the computer” on page 48

Workflows for Ethernet networking

About this topic

If you have not yet decided whether to use a serial or Ethernet connection to connect your device to the controlling computer, read "Choosing between serial and Ethernet connections" on page 51.

This topic guides you through the Ethernet connection setup process according to the type of network you want to set up.

Ethernet connection workflow

This is the workflow you use to set up an Ethernet connection.

Skip step 6 if you are assigning IP addresses statically.

Step	Topic
1	"About Ethernet networking" on page 52
2	"Setting the network card's IP address" on page 56
3	Setting the device's IP address Refer to the device documentation
4	"Connecting Ethernet cables" on page 59
5	"Checking the network card's IP address" on page 60 *Optional*. You can do this at any time.
6	*Dynamic IP addresses only*. "Installing and starting DHCP server software" on page 61
7	"Turning off Ethernet firewall software" on page 63

Setting the network card's IP address

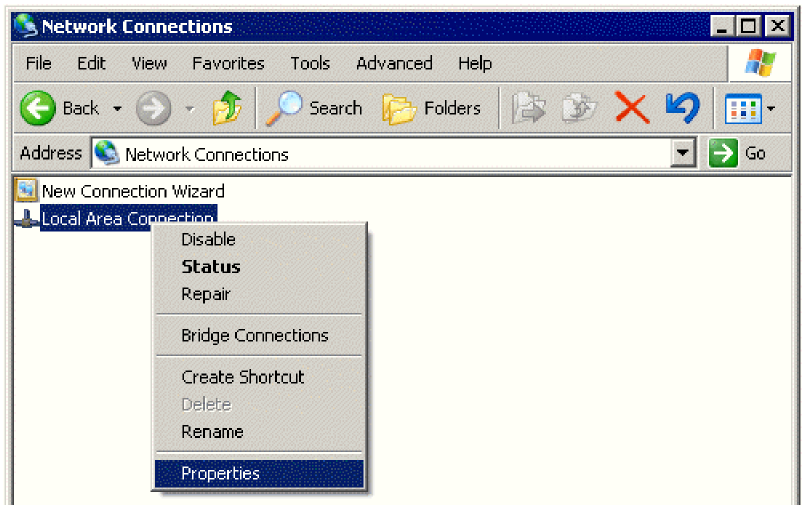
About this topic

This topic describes how to manually set a static IP address for an Ethernet connection between the controlling computer and a device.

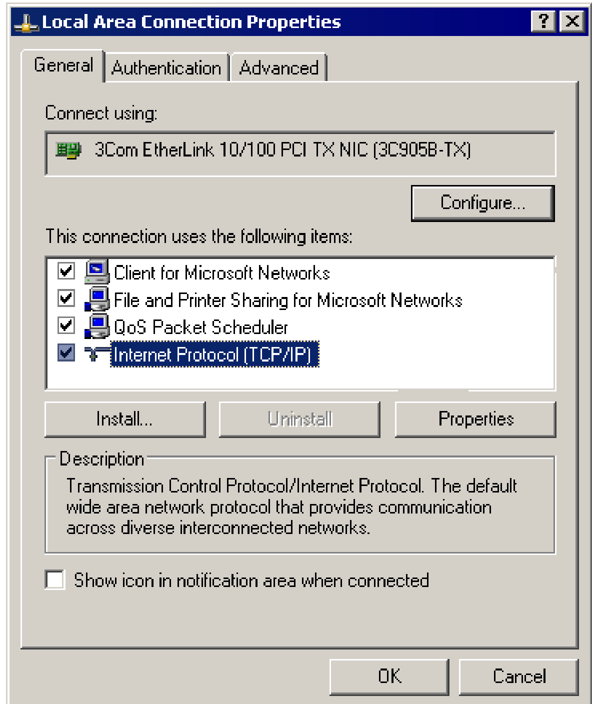
Procedure

To manually set the network card's IP address:

1. In Windows, select **Start** > **Settings** > **Network and Dial-up Connections**.
2. Right-click the icon for the new local area connection (which might be **Local Area Connection 2** on your system), and click **Properties**.



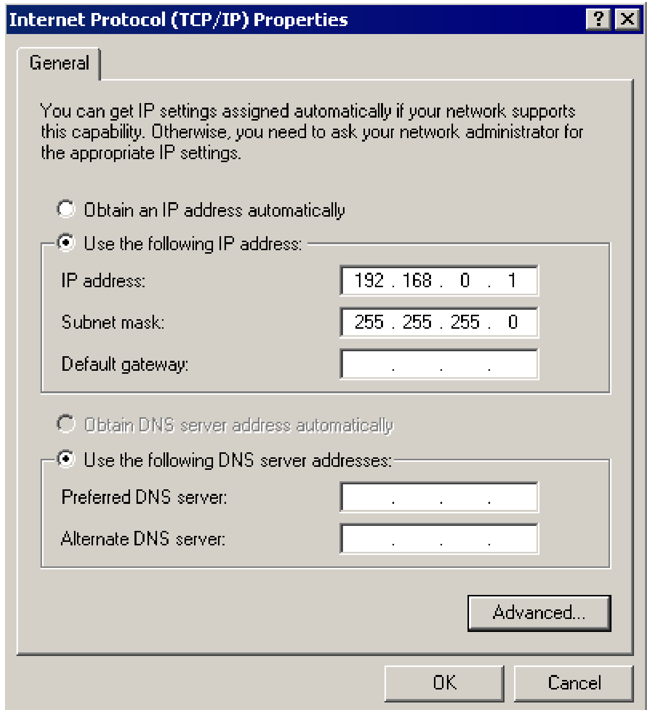
3. Double-click **Internet Protocol (TCP/IP)**.



4. Select **Use the Following IP address,** and enter the IP address and the subnet mask.

   If you are using the DHCP server software provided by Velocity11, set the IP address to `192.168.0.1` and the subnet mask to `255.255.255.0`

   This manually configures the IP address of the card.



5. Click **OK**.
6. Close the **Network Connections** window.

About setting devices' IP addresses

You also need to set the IP addresses for the devices on the network. This requires connecting to the devices with a serial cable and using HyperTerminal to change the IP addresses in the devices' firmware.

Details about how to do this should be provided by the manufacturer of the device.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflow for installing VWorks and networking the computer" on page 48
Installing the DHCP server software from Velocity11	"Installing and starting DHCP server software" on page 61

Connecting Ethernet cables

About this topic

This topic describes how to connect the Ethernet cable from the controlling computer to your instrument.

Ways to connect

You can use Ethernet to connect a device to the computer:

- ❑ Directly
- ❑ Through an Ethernet switch

If you are setting up a stand-alone device, you can use either method. If you are setting up a local area network that has other devices on it, follow only the directions for using a switch.

Before you start

Before starting this procedure, and if you are setting up a new network, make sure the IP address of the computer's network card has already been statically assigned.

To connect your device directly to the computer, you need a crossover Ethernet cable, which can be obtained from a computer supply retail outlet.

Connecting directly

To connect directly to the computer with Ethernet:

1. Connect a crossover Ethernet cable to the Ethernet port of the connections panel.
2. Connect the other end of the cable to the Ethernet port of the computer.

Connecting through a switch

An Ethernet switch connects the single cable from the computer to one or more cables that lead to one or more devices. A switch is included in the Velocity11 DHCP Server Kit.

Note: Do not use a crossover cable with a switch. The switch performs the crossover function.

If you are setting up a new network, connect the Ethernet switch to the computer and then connect the devices to the Ethernet switch using the procedures in this section.

If you are adding devices to an existing LAN, you do not need to perform the first procedure because the switch should already be connected.

To connect the switch to the computer:

1. Connect the power cord to the switch.
2. Plug one end of the Ethernet cable into any Ethernet port of the switch.
3. Plug the other end into the available Ethernet port of the computer.

To connect a device to the switch:

1. Plug one end of a second Ethernet cable into any available port of the Ethernet switch.
2. Plug the other end into the Ethernet port of the device.

Connecting additional devices

You can connect as many devices to the network as there are Ethernet ports available.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflows for Ethernet networking" on page 55
Ethernet networking	"About Ethernet networking" on page 52

Checking the network card's IP address

Introduction

When setting up an Ethernet network, you might want to verify the IP address of the network card in the computer. This topic describes how to do that.

Before you start

Before you can check the IP address of a computer, the computer's network card must have been manually assigned a static IP address.

Procedure

To check the IP address of a computer's network card:

1. On the computer whose IP address you want to check, in Windows, select **Start** > **Programs** or **All Programs** > **Accessories** > **Command Prompt**.

   A Command Prompt window opens.
2. Type `ipconfig` and press ENTER.

   One IP address for every network card in your computer is shown.

Related topics

For information about...	See...
Ethernet networks	"About Ethernet networking" on page 52
The workflow that this procedure belongs to	"Workflows for Ethernet networking" on page 55

Installing and starting DHCP server software

About this topic

Follow the procedure in this topic if you are using the DHCP server software provided by Velocity11 to set up your Ethernet local area network with dynamic IP addressing. Make sure that you have already connected your device to the computer.

You can choose to use DHCP server software from another source if you prefer.

What is a DHCP server

A DHCP server is a software program that dynamically assigns IP addresses to devices in a LAN when they are turned on. Once installed, the server runs as a Windows service that starts automatically when you start the computer.

The server software is included in the DHCP Server Kit from Velocity11.

!! IMPORTANT !! You should only run one DHCP server on one network, so do not install the Velocity11 DHCP server if you are integrating your device into an existing network that has dynamic IP address assignment.

Installing the DHCP software

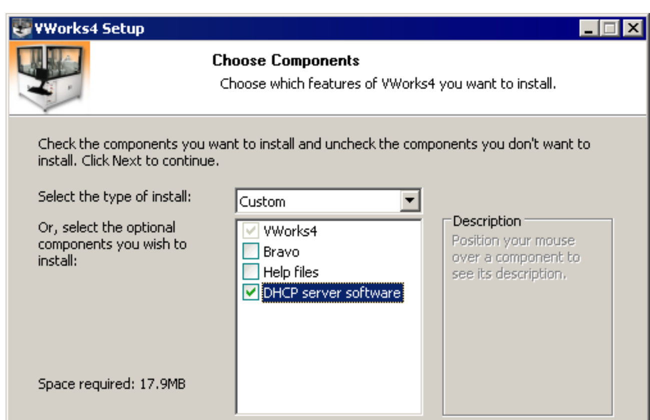
If you are using the computer that shipped with VWorks, the DHCP server software should already be on the computer (and you should skip to the next procedure).

Note: Because the DHCP software runs as a service, you might need additional privileges on the controlling computer to install it. Check with your network administrator if you think you do not have sufficient privileges.

To install the DHCP server software:

1. Insert the CD-ROM for the device (or VWorks) into the controlling computer's CD-ROM drive.
2. Double-click the setup.exe file.
3. Follow the instructions in the wizard to complete your installation.

   Make sure you select DHCP server software.

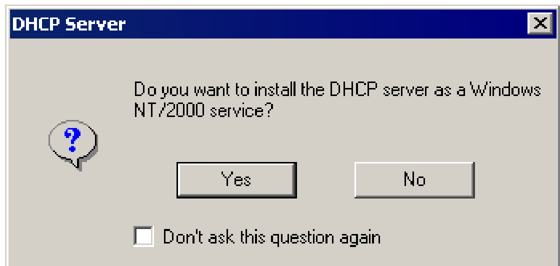


Starting the DCHP server software

To install and restart the DHCP server software:

1. Navigate to the folder where the DHCP server software is installed, for example, C:\Program Files\Velocity11\VWorks4\DHCP.
2. Double-click the file named dhcpsrv.exe.

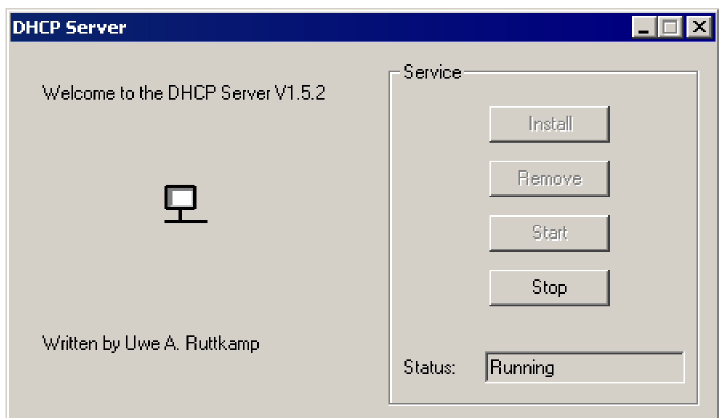
   The **DHCP Server** window opens.



3. Click **Yes.**

   This installs the DHCP server on your computer.

   The **DHCP Server** window opens and displays the status as **Running**.



4. Close the window.

Stopping and restarting the service

To stop and restart the DHCP server using the DHCP server file:

1. Navigate to C:\Program Files\Velocity11\VWorks4\DHCP.
2. Double-click the file named dhcpsrv.exe.
3. Click **Stop** to stop the service or **Start** to start the service.

Related topics

For information about...	See...
Ethernet networks	"About Ethernet networking" on page 52
The workflow that this procedure belongs to	"Workflows for Ethernet networking" on page 55

Turning off Ethernet firewall software

Introduction

If you are using an Ethernet network, read this topic to learn about the effect of firewalls and how to turn them off.

Firewall defined

A firewall is a security system that protects a computer on an Ethernet network against external threats.

Types of firewall

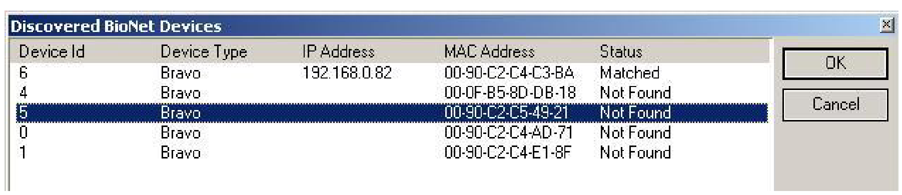
Hardware firewalls protect computers from threats outside the local area network (LAN). Software firewalls prevent worms from propagating on the internal network.

Sources of software firewall

The computer controlling your device might have one or more software firewalls running on it. There are two software firewall sources to be concerned with:

- ❑ The Windows XP operating system has a built-in firewall that might be turned on by default or might have been turned on by someone in your organization.
- ❑ Your computer might be running a separate firewall program that your organization has purchased and installed.

Either of these types of firewall can prevent the devices on the network from showing up in the Discovered BioNet Devices dialog box.



With dynamic IP addressing

If your Ethernet network is using dynamic IP addressing, the firewall might block the request for an IP address from your device when it is turned on.

With static IP addressing

If your Ethernet network is using static IP addressing, the VWorks software sends out a request for devices on the network to identify themselves, but the firewall might block the response.

About turning off firewalls

To turn off the firewall for your operating system, follow the directions in Windows Help and Support on your computer.

To turn off another firewall program, follow the directions in the documentation that accompanies the program. You can either uninstall the program or stop it. If you choose to stop it, make sure that the program is not configured to start up again every time you restart the computer.

Passing data through the firewall

If your organization insists that a firewall should remain on, it is possible to configure the firewall to allow data that is transmitted on a particular port to pass through.

If you are using dynamic IP addresses, the firewall should pass data though on UDP ports 67 and 7611.

If you are using static IP addresses, the firewall should pass data through on UDP port 7611.

For help configuring the firewall, consult your network administrator.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflows for Ethernet networking" on page 55

Uninstalling VWorks

About this topic

This topic describes how to uninstall VWorks.

When to remove Velocity11 registry files

In general, it is sufficient to uninstall the VWorks program without removing the registry files. However, you can remove the Velocity11 files from the registry if:

- ❑ You want to make a completely fresh start with VWorks, removing all user accounts, teachpoints, device profiles, and liquid and labware definitions, or
- ❑ You do not intend to run VWorks on that computer again

Procedures

To remove VWorks:

1. Use the Add / Remove Programs control panel.

   For more information, see the online help for your Windows operating system.

   !! IMPORTANT !! The following procedure deletes the user accounts, labware definitions, liquid library data, device profiles, and teachpoints.

To remove the Velocity11 files from the registry:

1. From the Windows **Start** menu, select **Run**.
2. In the **Open** text box, type `regedit`.
3. Click **OK**.

   The Windows registry editor opens.
4. Expand folders to select the following folder:

   HKEY_LOCAL_MACHINE\SOFTWARE\Velocity11
5. Make sure you have selected the Velocity11 folder.

   !! IMPORTANT !! Making a mistake and deleting the wrong registry folder may cause critical failures with your operating system.
6. Select **Edit** > **Delete**.

Related topics

For information about...	See...
Types of information stored in the registry	“Relationships of VWorks components” on page 24
Exporting Velocity11 data from the registry files (for example, before clearing the registry)	“Moving or sending a registry file” on page 183

Creating a protocol

6

The procedures in this chapter can be performed by people with administrator or technician privileges. This chapter provides the basic reference information you need to write protocols.

This chapter contains the following topics:

- ❑ "About tasks, processes, and protocols" on page 68
- ❑ "About protocol files" on page 71
- ❑ "An example protocol" on page 72
- ❑ "Workflow for creating a protocol" on page 73
- ❑ "Creating a new protocol" on page 74
- ❑ "Setting protocol options" on page 75
- ❑ "Setting the plate properties for a protocol process" on page 77
- ❑ "About setting the number of simultaneous plates" on page 79
- ❑ "Setting up a startup or cleanup protocol process" on page 80
- ❑ "Adding and deleting tasks" on page 81
- ❑ "Compiling protocols" on page 83
- ❑ "Saving protocols" on page 84
- ❑ "Simulating a run" on page 84
- ❑ "Tracking plates using bar codes" on page 86

About tasks, processes, and protocols

About this topic

This topic defines some terms that you need to know before you can understand or create protocols.

Plate process defined

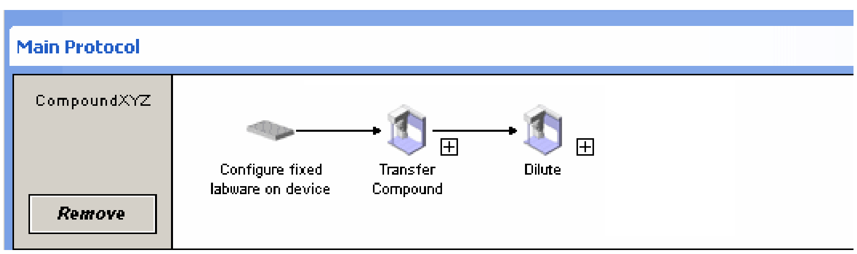
When multiple processes are permitted, each process thread starts with a plate process icon. This icon represents the basic information about a plate or collection of plates and is always the first icon in a process displayed in the protocol editor. It has associated parameters that are defined in the Task Parameters toolbar.

Task defined

A task is an operation usually performed by a device, on one or more plates, and is represented by an icon in the protocol editor. It has associated parameters that are defined in the Task Parameters toolbar.

The following example icon represents a task for upstacking plates.

The following example shows a process with one plate icon followed by two task icons.

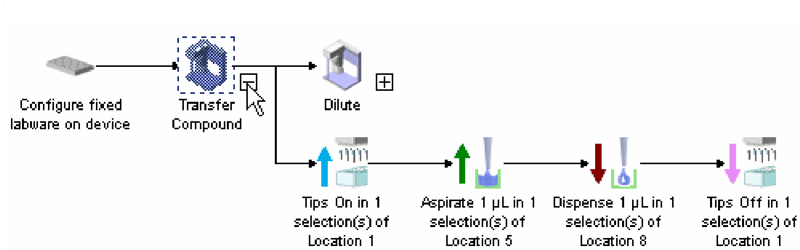


Sub-process defined

A sub-process is a sequence of tasks performed as a sub-routine within a protocol. Sub-processes are represented by a sub-process icon in the protocol.

Typically, pipetting tasks can only be added to a protocol process as part of a sub-process. The sub-process icon can be expanded or collapsed to make a process easier to read.

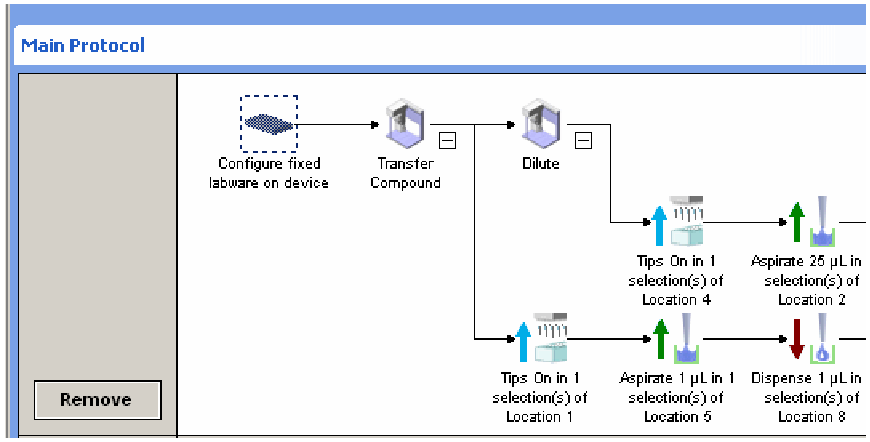
The following diagram shows an expanded sub-process.



Protocol defined

A protocol is one or more processes that are run together.

The following example shows one protocol with two sub-processes.



Startup protocol defined

A startup protocol is one or more processes that are executed once, before the protocol. The processes are created in the startup protocol editor which is accessed by clicking the Startup tab in the protocol workspace.

Priming reservoir pumps is an example of a startup-protocol task.

When you click Start, if there is a startup protocol, it is executed first, followed by the protocol.

Cleanup protocol defined

A cleanup protocol is one or more processes, which are executed once after the protocol. The processes are created in the Cleanup protocol editor which is accessed by clicking the Cleanup tab in the protocol workspace.

A cleanup protocol is typically used to clean the system after a protocol run.

When you click Start, a startup protocol might run first, followed by the protocol, and then any cleanup protocol processes.

Run defined

A run is a single protocol, which includes any startup protocol and cleanup protocol processes, that is performed one or more times in a series.

When you start a run, you are prompted to enter the number of cycles that you want to run. The value that you enter represents the number of times the protocol executes before the run ends. For example, a value of four means that the protocol runs four times in the series. Note that when you run a protocol multiple times, the startup and cleanup, if they exist, are only run once.

Related topics

For information about...	See...
Workflow for creating a protocol	"Workflow for creating a protocol" on page 73
Protocol file format	❑ "About protocol files" on page 71 ❑ "An example protocol" on page 72
Creating a protocol	"Creating a new protocol" on page 74

About protocol files

About this topic

This topic provides an overview of the protocol file format.

File format

Protocols are created in the drag-and-drop protocol editor. When they are saved, the information is written to a file in XML format. In XML, the elements indicate the protocol's properties, and text within the markup tags gives the properties' values.

You can create and edit protocols in the protocol editor or directly in XML. The XML files can also be useful for troubleshooting because you can, for example, see which device file is associated with the protocol.

XML example

A protocol file viewed as XML is shown below. You can open a protocol file in any browser that contains an XML parser, for example, Mozilla Firefox.

```
  </Task>
- <Task Name="Bravo::secondary::Aspirate" Task_Type="0">
  - <Advanced_Settings>
      <Setting Name="Estimated time" Value="5.0"/>
      <Setting Name="Task has timing constraint" Value="0"/>
      <Setting Name="Minimum time" Value=""/>
      <Setting Name="Maximum time" Value=""/>
    </Advanced_Settings>
  - <Parameters>
      <Parameter Name="Location" Value="7"/>
      <Parameter Name="Volume" Value="10"/>
      <Parameter Name="Pre-aspirate volume" Value="0"/>
      <Parameter Name="Post-aspirate volume" Value="0"/>
      <Parameter Name="Liquid class" Value="< None >"/>
      <Parameter Name="Distance from well bottom" Value="2"/>
      <Parameter Name="Dynamic tip extension" Value="0"/>
      <Parameter Name="Number of sides to tip touch" Value="0"/>
      <Parameter Name="Tip touch retract distance" Value="0"/>
      <Parameter Name="Tip touch horizontal offset" Value="0"/>
      <Parameter Name="Well selection" Value="<?xml version='1.0' encoding='ASCII' ?> <Velocity11 file='M
      <Channels Value='0' /> <Quadrant Column='0' Row='0' /> </Velocity11>"/>
    </Parameters>
    <PipetteHead Channels="0" Disposable="1" MaxRange="245" MinRange="-40" Name="96LT, 200 �I
  </Task>
- <Devices>
    <Device Device_Name="Bravo - 1" Location_Name="1"/>
  </Devices>
```

Related topics

For information about...	See...
Tasks, processes, and protocols	"About tasks, processes, and protocols" on page 68
Protocol examples	"An example protocol" on page 72
Creating a protocol	"Workflow for creating a protocol" on page 73

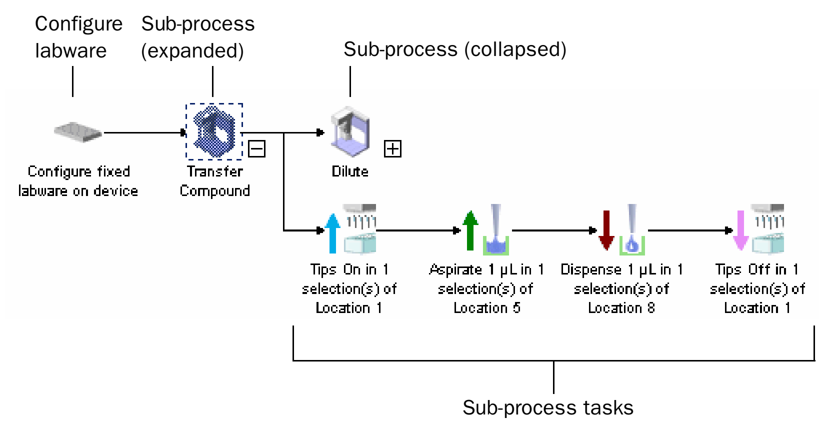
An example protocol

About this topic

This topic shows an example VWorks protocol for the Bravo.

Example protocol

This example protocol contains two VWorks sub-processes. The first one is shown expanded. It transfers 1 µL from the plate on location 5 to the plate on location 8.



Related topics

For information about...	See...
Understanding tasks, processes, and protocols	"About tasks, processes, and protocols" on page 68
Workflow for creating a protocol	"Workflow for creating a protocol" on page 73

Workflow for creating a protocol

About this topic

This topic outlines the procedures you need to follow to create a protocol in VWorks.

Workflow

The general workflow for creating a basic protocol is:

Step	Topic
1	"Creating a new protocol" on page 74
2	"Setting protocol options" on page 75
3	Creating a protocol process, which includes ❑ "Setting the plate properties for a protocol process" on page 77 ❑ "Setting up a startup or cleanup protocol process" on page 80 ❑ "Adding and deleting tasks" on page 81
4	"Compiling protocols" on page 83
5	"Saving protocols" on page 84
6	"Simulating a run" on page 84
7	"Tracking plates using bar codes" on page 86

Related topics

For information about...	See...
Tasks, processes, and protocols	"About tasks, processes, and protocols" on page 68
Protocol file format	"About protocol files" on page 71
How many plates to have in the system at once	"About setting the number of simultaneous plates" on page 79

Creating a new protocol

About this topic

This topic describes how to create a new protocol.

Procedure

To create a new protocol:

1. Open VWorks.
2. Select **File** > **New** > **Protocol**.

Related topics

For information about…	See…
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 73
Associate a device file with a protocol	"Setting protocol options" on page 75
Open an existing protocol	"Opening a protocol in VWorks" on page 31
Moving between existing protocols	"Opening a protocol in VWorks" on page 31
What to do next	"Setting protocol options" on page 75

Setting protocol options

About this topic

When you create a protocol, you need to specify which device file to use. Optionally, you can specify additional information to associate with your protocol file.

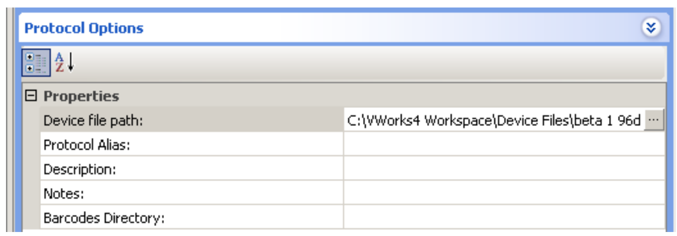
This topic describes how to choose the device file and add additional information.

Choosing the device file for a protocol

To choose the device file for the protocol:

1. Click the **Protocol Options** tab.
2. Double-click the field adjacent to **Device file path** to open the **Open** file dialog box.
3. Select the device file (.dev file type) and click **Open**.

   The file path to the device file is displayed in the field.



Adding information about the protocol

You can associate the following with your protocol:

- ❑ A Protocol Alias (short name) to appear in the Workspace list
- ❑ A Description and Notes

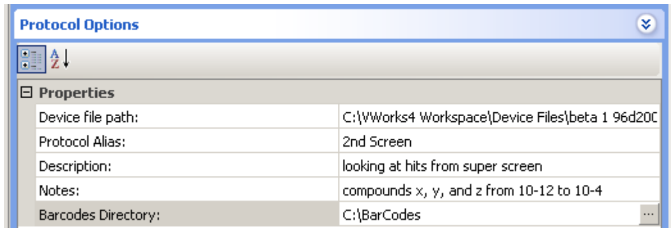
  Use this text to remind yourself or others of details about this protocol.
- ❑ A Bar codes Directory to be available in the Plate Properties of the protocol process

To add notes about your protocol:

1. Click the **Protocol Options** tab.
2. Enter values for the **Properties**.

To associate a bar code directory:

1. Click the **Protocol Options** tab.
2. Click the ellipsis button and select the bar code folder from the **Browse for folder** dialog box.
3. Click **OK**.



Related topics

For information about...	See...
Compiling and saving protocols	"Compiling protocols" on page 83
Workflow this procedure belongs to	"Workflow for creating a protocol" on page 73
What to do next	"Setting the plate properties for a protocol process" on page 77
Adding tasks to protocols	"Adding and deleting tasks" on page 81
Setting up bar code tracking	"Tracking plates using bar codes" on page 86

Setting the plate properties for a protocol process

About this topic

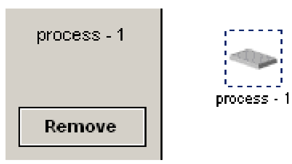
This topic describes setting the plate properties for a protocol process. This step is only required if you are running VWorks Benchtop.

Procedure

To set the plate properties for a protocol process:

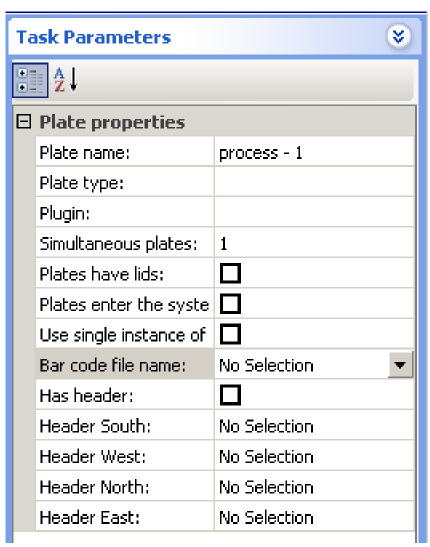
1. In the protocol editor, click **Add**.

A plate icon appears in the protocol editor window.



2. In the **Task Parameters** toolbar, set the **Plate properties**.

Use the table below as a guide.



Property	Description
Plate name	Enter a name to help you identify the plate process. For example, Source or Destination.
Plate type	Select a plate type from the list. This list is created in the labware editor.

Property	Description
Plugin	This property is not implemented in this version of VWorks
Simultaneous plates	Set to 1 if you are running VWorks Instrument license.
Plates have lids	This property is not implemented in this version of VWorks
Plate enter the system sealed	This property is not implemented in this version of VWorks
Use single instance of plate?	Not implemented for this version of VWorks.
Barcode File Name	Select the file from the list if you have a bar code file to associate with this process.
Has Header	Select if the first line of the bar code file name has a header.
Header South	For the South side of the plate, select the appropriate column.
Header West	For the West side of the plate, select the appropriate column.
Header North	For the North side of the plate, select the appropriate column.
Header East	For the East side of the plate, select the appropriate column.

Related topics

For information about...	See...
How to set the number of simultaneous plates	"About setting the number of simultaneous plates" on page 79
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 73
What to do next	"Setting up a startup or cleanup protocol process" on page 80
Reading bar codes from a file	"Tracking plates using bar codes" on page 86

About setting the number of simultaneous plates

About this topic

This topic provides guidance on how to set the number of simultaneous plates in the Task Parameters. (See “Setting the plate properties for a protocol process” on page 77.)

Selecting the number

The number of simultaneous plates is the maximum number of plates belonging to a single process that are in the system at one time.

Plates that are in the system could be in the robot gripper, on platepads, PlateLoc, and so on.

The number of simultaneous plates to select for a plate process depends on how many tasks there are in the protocol.

In general, one simultaneous plate can be used for every task in the protocol. This is because, in general, each task uses one plate position. As an example, a protocol with three tasks can have three plates on the table at once. There are exceptions to this, though, such as cases where the same plate position is used for more than one of the tasks and when a VPrep is used. Several plates can be positioned on a VPrep at the same time.

If the number of simultaneous plates is set too high, the protocol might be slowed down because the robot will move around to avoid a deadlock. In cases where too many plates are in the system and there is no way to move plates around further, this can result in a deadlock, the point at which the protocol stops.

If the number of simultaneous plates is set too low, the time for the protocol run could be extended. Optimizing the number of simultaneous plates is therefore critical for maximizing efficiency.

A typical protocol has 5–30 simultaneous plates, but it could be many more than that. A safe number to start with when testing a new protocol is 1–2. Run the protocol in simulation to optimize the number of simultaneous plates.

Related topics

For information about...	See...
The place where the number of simultaneous plates is entered	“Setting the plate properties for a protocol process” on page 77

Setting up a startup or cleanup protocol process

About this topic

This topic describes how to set up a Startup or Cleanup protocol process. These processes are available only when you are running VWorks Benchtop.

Startup protocols are processes that are carried out once, before the protocol is executed. Cleanup protocols are processes that are carried out once, after the protocol is executed.

When to use

Use the startup protocol editor when you want a task performed one time before the protocol runs. For example, if you are dispensing reagents, you might want to prime a pump with fluid.

Use the cleanup protocol editor when you want to perform a task one time after the protocol runs. For example, you might want to purge lines with a buffer or cleaning agent.

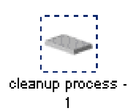
Procedure

To set up a startup protocol or cleanup protocol process:

1. Click the **Startup Protocol** or the **Cleanup Protocol** tab.
2. Click **Add**.

   A startup protocol process icon appears in the **Startup Protocol Editor** or the **Cleanup Protocol Editor** window.

3. Type in a name for the process in the **Process name** field.
4. Create the startup protocol process as you would a protocol process, by adding tasks and then setting the task parameters.
5. When you are finished, compile and check your startup protocol or cleanup protocol.

Related topics

For information about...	See...
Workflow this procedure belongs to	“Workflow for creating a protocol” on page 73
Adding and deleting tasks	“Adding and deleting tasks” on page 81
Compiling a protocol	“Compiling protocols” on page 83

For information about...	See...
Running a protocol in simulation mode	"Simulating a run" on page 84

Adding and deleting tasks

About this topic

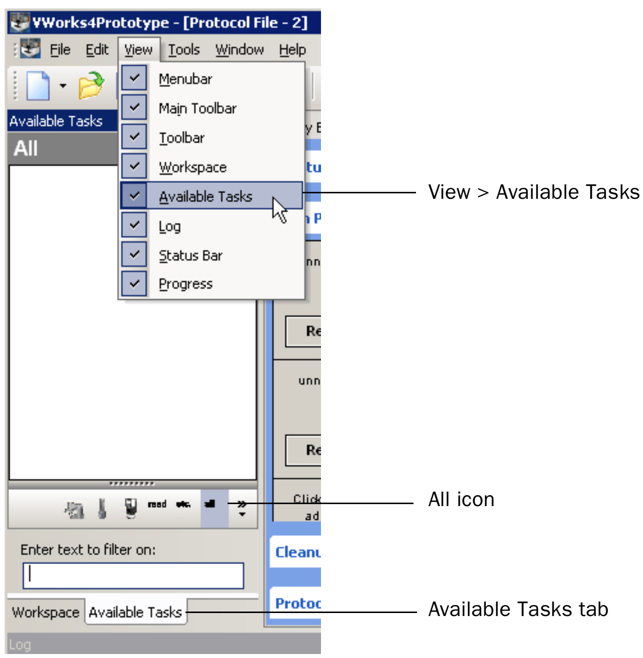
After you have set up a plate icon for a process, you can start adding tasks. This topic describes how to do this.

You must always add a task to a process before you can define its task parameters.

Adding a task

To add a task to a process:

1. If the Available Tasks are not visible, make sure that:
   a. **View** > **Available Tasks** is selected.
   b. The **Available Tasks** tab is selected (not **Workspace**).
   c. The **All** icon is selected.



2. To add tasks to the protocol, do one of the following:
   - Click a task icon and drag it from the task list to the protocol editor window until a vertical, dashed line appears.
   - Double-click the icon.
   - Copy (or cut) and paste task icons in the protocol.

Deleting a task

To delete a task from a process:

1. In a protocol editor, select a task that is in a protocol process.
2. Press the DELETE key on the keyboard.
3. Click **Yes** in the **Delete Task** dialog box to delete the task.

Moving tasks

When editing a protocol, drag-and-drop or use cut-and-paste commands to move tasks and groups of tasks.

To move tasks in a protocol:

1. In a protocol editor, select a task or a group of tasks in a protocol process.
2. Do one of the following:
   - Drag and drop the tasks to a new location in the protocol.
   - Use the **Cut**, or **Copy** and **Paste** commands on the **Edit** menu to move the tasks. To paste between two tasks, click the first task before pasting the copied task.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 73
What to do next	"Compiling protocols" on page 83
Setting Bravo task parameters	"List of Bravo task parameters" on page 90

Compiling protocols

About this topic

This topic describes how to compile a protocol.

Compiling a protocol

To compile a protocol:

1. Click **Compile**.

   Errors are reported in the Log toolbar.

   Whenever you start a protocol, VWorks automatically compiles it and checks for errors such as:

   - Pipette volumes
   - Tips required for a process
   - Labware class prohibited from a location

Related topics

For information about...	See...
The workflow that this procedure belongs to	“Workflow for creating a protocol” on page 73
Troubleshooting compiling errors	“Compilation warnings and errors” on page 187
Saving the protocol	“Saving protocols” on page 84

Saving protocols

About this topic

This topic describes how to save a protocol.

Saving a protocol

To save a protocol you must be logged in with an administrator or technician user account.

!! IMPORTANT !! When you edit a protocol, the changes take effect immediately. However, unless you explicitly save the protocol, the changes are lost when you exit VWorks.

To save a protocol:

1. Select **File > Save As**.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 191
What to do next	"Simulating a run" on page 215

Simulating a run

About this topic

This topic describes how to use simulation run to check for errors in a protocol after it is compiled.

Simulating the run

After making sure there are no compiler errors in the protocol, you can check for other types of problems by running the protocol through the simulator. The simulator allows you to confirm that steps are completed and sequenced correctly and to find problems such as:

- ❑ Deadlocks
- ❑ Periods of inefficiency, such as when the robot is not being used
- ❑ Plates spending different times at critical steps when they should be run under identical conditions
- ❑ A number of simultaneous plates that is too high or too low

The simulator does not move plates. It performs a virtual run based on the estimated execution times displayed for each task. You can change the execution times for tasks to make the simulation more accurate for your protocol.

One approach you can use for testing is the following:

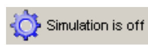
1. Run the simulator with the default task execution times and the same number of plates expected for a run to identify deadlocks and rate-limiting tasks.

2. Resolve any major problems with the protocol.
3. Perform a real dry run with a plate.
4. Use the times recorded in the Log toolbar to edit the task execution times for each task.
5. Run the simulator with the more accurate task execution times.
6. Fine-tune the protocol based on the results of the simulation.

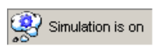
Running the simulator

To run the simulator:

1. Click **Simulation is off** on the toolbar.

   The button changes to **Simulation is on**.

2. Click **Start** to run the protocol in simulation mode.

Related topics

For information about...	See...
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 191
What to do next	"Tracking plates using bar codes" on page 86

Tracking plates using bar codes

About this topic

This topic describes how to track plates with a bar code file using VWorks.

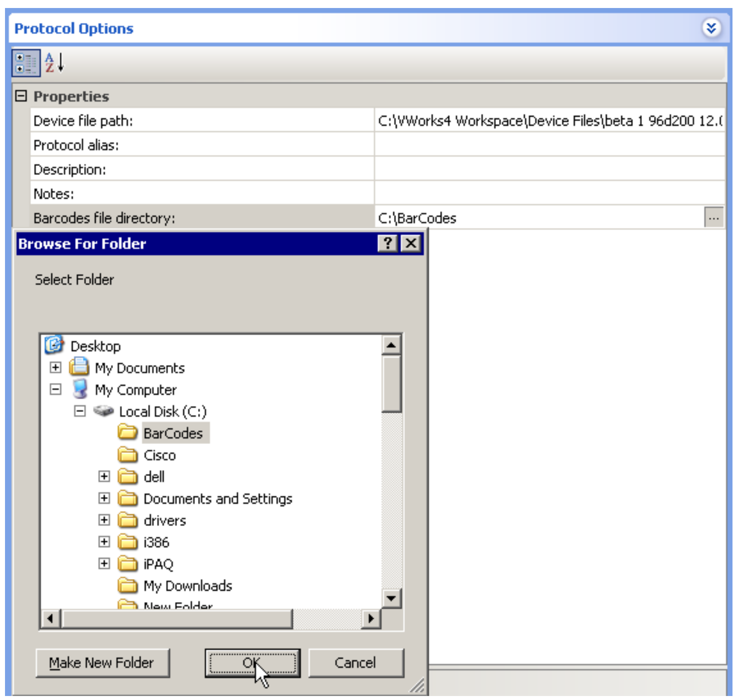
Before you start

Before you start, create a tab-delimited or comma-separated file, in the .txt format, containing the bar codes for the plates you are running.

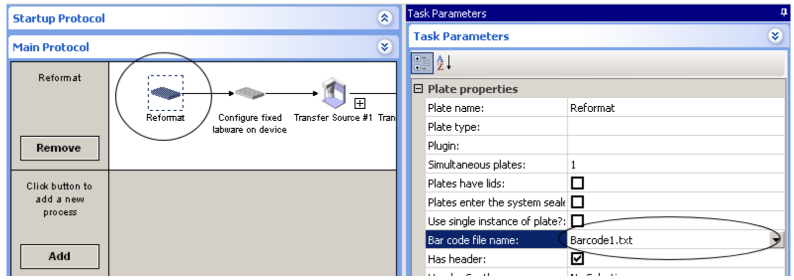
Procedure

To set up bar code tracking in VWorks:

1. Associate the bar code folder with the protocol:
   a. Click the **Protocol Options** tab.
   b. Click the ellipsis button adjacent to Barcodes **file directory**.
   c. Select the folder containing the bar codes.
   d. Click **OK**.



2. Associate the protocol process with a specific bar code file:
   a. Click the **Main Protocol** tab.
   b. Select the plate process icon.
   c. In the **Task Parameters** toolbar, select the bar code file to use.



d. If the file has a header, select **Has Header**.

   When **Has Header** is checked, the software will ignore the information in the first row of the spreadsheet file. A header contains information about each column of barcode, such as the side of the plate (N,S,W,E) that corresponds to a column, but does not contain any barcode data.

e. For each side of the plate, select the appropriate column.

   Each process may have four columns and a maximum of one column per side.

3. Repeat steps 1–2 for each process in the protocol so that each process is associated with a bar code file.

4. Direct VWorks to the first plate in the bar code file:

   a. When you are ready to start the run, click **Start**.

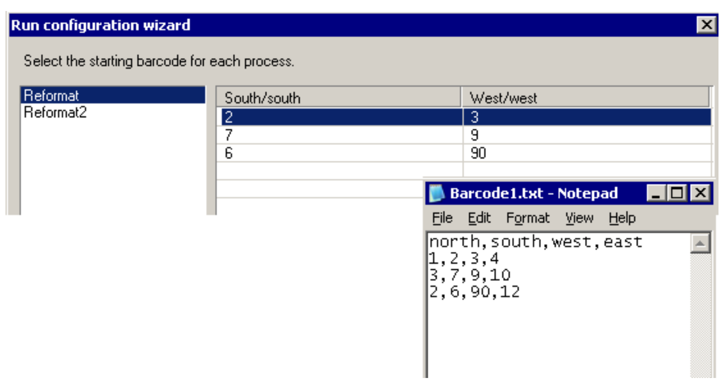
      This starts the **Run configuration wizard**.

   b. Follow the directions in the first step of the wizard.

   c. Click **Next**.

   d. Select a process on the left and then select the bar code line on the right that is associated with the first plate of that process.

      The screen below shows an example accompanied by the corresponding .txt file for that process.



   e. Repeat for each process in the protocol.

   f. Click **Finish**.

Note: When the plate is processed, the bar code is displayed in the log.

Related topics

For information about...	See...
Workflow this topic belongs to	"Workflow for creating a protocol" on page 73
Setting protocol options	"Setting protocol options" on page 75

Setting Bravo task parameters

7

This chapter explains how to write protocols for the Bravo.

This chapter contains the following topics:

- "List of Bravo task parameters" on page 90
- "Setting Configure Labware (Built In) task parameters" on page 92
- "Setting location properties for Bravo pipette tasks" on page 94
- "Setting Aspirate (Bravo) task parameters" on page 96
- "Setting Dispense (Bravo) task parameters" on page 99
- "Setting Mix (Bravo) task parameters" on page 102
- "Setting Move to Location (Bravo) task parameters" on page 105
- "Setting Pump Reagent (Bravo) task parameters" on page 106
- "Setting the Reserve Location task parameters" on page 108
- "Setting Serial Dilution (Bravo) task parameters" on page 109
- "Setting Set Head Mode (Bravo) task parameters" on page 117
- "Setting Shake (Bravo) task parameters" on page 118
- "Setting Sub Process (Bravo) task parameters" on page 119
- "Setting Tips On (Bravo) task parameters" on page 120
- "Setting Tips Off (Bravo) task parameters" on page 121
- "Setting User Message task parameters" on page 123
- "Setting Vacuum filter task parameters" on page 124
- "Setting Wait For User (Bravo) task parameters" on page 126
- "Setting Wash Tips (Bravo) task parameters" on page 127
- "Setting Device Selection in task parameters" on page 131
- "Setting Advanced Settings task parameters" on page 132
- "Using JavaScript in VWorks" on page 134
- "JavaScripts task object and properties" on page 136

List of Bravo task parameters

About this topic

This topic lists the task parameters for the Bravo and points you to topics that describe what they do and how to set their parameters.

Task parameters defined

The parameters specify the details of the task. After you have added a task, you can set parameters for it. Most tasks require you to set parameters. As you set the parameters, the text underneath the task icons change to reflect the new parameters.

Related information

The tasks that are available for the Bravo are listed in the following table and described later in this chapter.

Task	Use the task in this type of process ...	See ...
Aspirate	Bravo sub-process	"Setting Aspirate (Bravo) task parameters" on page 96
Configure Labware	Main process	"Setting Configure Labware (Built In) task parameters" on page 92
Dispense	Bravo sub-process	"Setting Dispense (Bravo) task parameters" on page 99
Javascript Task	All processes	❑ "Using JavaScript in VWorks" on page 134 ❑ "JavaScripts task object and properties" on page 136
Mix	Bravo sub-process	"Setting Mix (Bravo) task parameters" on page 102
Move To Location	❑ Bravo sub-process ❑ Startup process ❑ Cleanup process	"Setting Move to Location (Bravo) task parameters" on page 105
Pump Reagent	Bravo sub-process	"Setting Pump Reagent (Bravo) task parameters" on page 106
Reserve Location	Bravo sub-process	"Setting the Reserve Location task parameters" on page 108
Serial Dilution	Bravo sub-process	"Setting Serial Dilution (Bravo) task parameters" on page 109
Set Head Mode	Bravo sub-process	"Setting Set Head Mode (Bravo) task parameters" on page 117
Shake	Bravo sub-process	"Setting Shake (Bravo) task parameters" on page 118

Task	Use the task in this type of process ...	See ...
Sub Process	Main process	“Setting Sub Process (Bravo) task parameters” on page 119
Tips Off	Bravo sub-process	“Setting Tips Off (Bravo) task parameters” on page 121
Tips On	Bravo sub-process	“Setting Tips On (Bravo) task parameters” on page 120
User Message	All processes	“Setting User Message task parameters” on page 123
Vacuum filter	Bravo sub-process	“Setting Vacuum filter task parameters” on page 124
Wait For User	Bravo sub-process	“Setting Wait For User (Bravo) task parameters” on page 126
Wash Tips	Bravo sub-process	“Setting Wash Tips (Bravo) task parameters” on page 127

Related topics

For information about...	See...
Creating protocols	“Workflow for creating a protocol” on page 73
Location properties for pipetting tasks	“Setting location properties for Bravo pipette tasks” on page 94
Protocols	❑ “About protocol files” on page 71 ❑ “About tasks, processes, and protocols” on page 68

Setting Configure Labware (Built In) task parameters

About this topic

This topic describes how to set the parameters for the Configure Labware (Built In) task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit protocols

Configure Labware (Built In) task defined

Use this task to tell VWorks what type of labware is at each location of the Bravo's deck.

When to use

If you are running VWorks Instrument, you will have to add this as a task before you do any pipetting or tips task with the pipet head. Typically, this will be the first task.

If you are running VWorks Benchtop and you have multiple processes in your protocol, typically, you would use this in only one (and usually the first) process.

!! DAMAGE HAZARD !! Using the Configure Labware task in multiple processes might result in collisions of the robot gripper and labware.

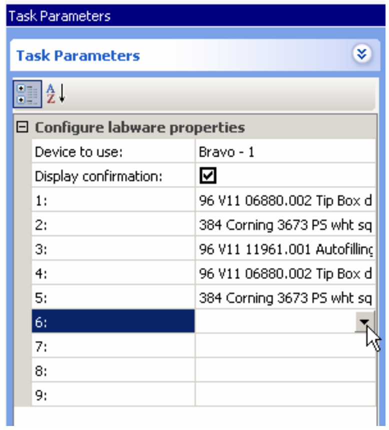
Procedure

To set the Configure Labware (Built In) task parameters:

1. Add the **Configure Labware (Built In)** task to a process in a protocol.
2. Set the **Configure labware properties**.

Property	Description
Device to use	Select the Bravo.
1: to 9:	Specify the labware that is on this location by selecting from the list.

3. Select **Display Confirmation** if you want to confirm labware placement before the run is started.



Related topics

For information about...	See...
Confirming labware before the run	"Starting a run from VWorks" on page 39
Creating protocols	"Workflow for creating a protocol" on page 73
Setting up labware types	"About defining labware with the labware editor" on page 140

Setting location properties for Bravo pipette tasks

About this topic

This topic describes what the location properties do and how they function in both licensing modes of VWorks. Read this topic if you are:

- An administrator or technician who writes protocols
- An operator who needs to edit the pipette tasks in a protocol

Location properties defined

Whenever you add a pipetting task to a protocol process, the location properties need to be set. There are two types:

- *Location, plate*—Setting this property tells VWorks which plate or plate process is receiving the action of the task
- *Location, location*—Setting this property tells VWorks at which location the task will occur.

Setting location properties with a VWorks Instrument license

With a VWorks Instrument license, you can only have one process in a protocol. Every task will occur on the Bravo deck and no plates will change location.

When creating a protocol, you will need to add a Configure Labware task which informs VWorks what labware is at each location. In this case, you will only need to set the Location, plate property because all deck locations have their labware defined.

Leave the Location, location property set to <auto-select>. With VWorks Instrument, the Location, location property is ignored.

Setting location properties with a Benchtop license

With a VWorks Benchtop license, you can have one or more processes in a protocol. You might not have labware at fixed positions and therefore might not want to use the Configure Labware task.

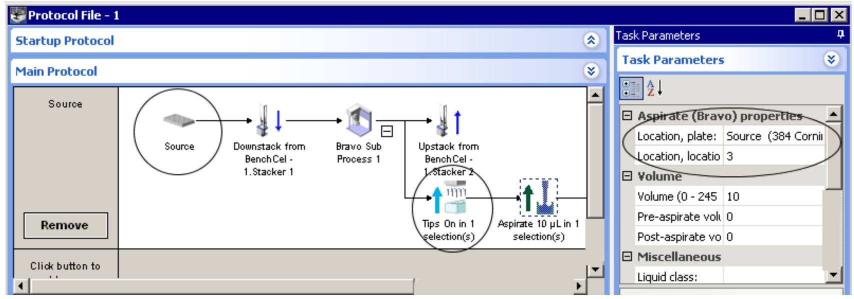
Because each process begins with a defined labware, the process itself can be set as the Location, plate. The Location, location is then set to the position on the Bravo where the task will take place.

In the example below, the protocol is aspirating 10 µL from a downstacked plate. The Location, plate is set to the process *Source*, which is a 384-well plate downstacked from a BenchCel. The location of the plate that will be aspirated, set in Location, location, will be location 3 on the Bravo.

Using <auto-select>

You can also set the Location, location property to <auto-select>. When set to <auto-select>, the robot will choose an appropriate location.

!! DAMAGE HAZARD !! Avoid potential collisions. When using this feature with Bravo accessories, make sure that you assign the correct allowed/prohibited labware classes in the location properties of the device file.



Related topics

For information about...	See...
Tasks used with Bravo	"List of Bravo task parameters" on page 90
Setting location properties for Bravo	*Bravo User Guide*
Protocol basics	"About tasks, processes, and protocols" on page 68
Setting a sub-process task	"Setting Sub Process (Bravo) task parameters" on page 119

Setting Aspirate (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Aspirate task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Aspirate task in a protocol

Aspirate task defined

An Aspirate task draws liquid from a plate or reservoir. This task is available in Bravo sub-processes.

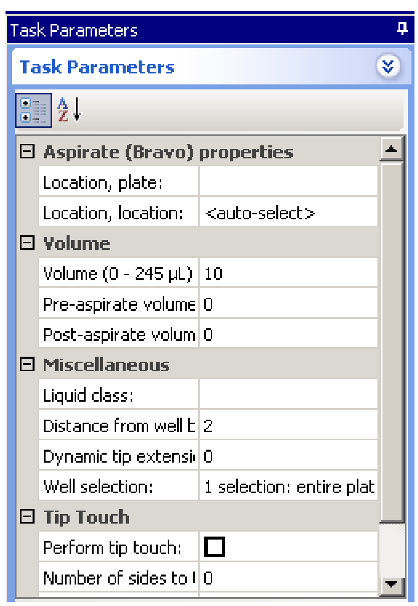
Procedure

To set the Aspirate (Bravo) task parameters:

1. Add the **Aspirate (Bravo)** task to a sub-process in a protocol.
2. Set the **Aspirate (Bravo) properties**.

Property	Description
Location, plate	Identifies the plate at which the aspiration will occur.
Location, location	Identifies the location at which the aspiration will occur. If you select autoselect, then the robot will choose an available location. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.
Volume	Specifies the volume of liquid to be drawn into each pipette tip.
Pre-aspirate volume	Specifies the volume of air to be drawn before the pipette tips enter the plate.
Post-aspirate volume	Specifies the volume of air to be drawn after the liquid is drawn up.
Liquid class	Indicates you have defined a liquid class for this liquid.
Distance from well bottom (0–100 mm)	Specifies the starting or maximum distance from the well bottoms that the tips will be during the aspirate cycle.
Dynamic tip extension (0–20 mm/µL)	Specifies the distance (in millimeters) to lower the head for each microliter aspirated.

Property	Description
Well selection	Identifies the wells for aspiration. Applies only if the Bravo head has fewer tips than the plate has wells or if you are in single row/column mode. Click the ellipsis button, and, in the Well Selection dialog box, click wells, rows, or columns to highlight the wells for aspiration.
Perform tip touch	Specifies whether a tip touch is performed after each selection of the plate.
Number of sides to tip touch	Specifies the number of places on the side of the well that the tip touches after aspiration. The number ranges from 0 to 4.
Tip touch retract distance (-20–20 mm)	Specifies the height that the tips move up before touching the sides of the wells.
Tip touch horizontal offset (-5–5 mm)	When the value for this parameter is 0, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.



Related topics

For information about...	See...
Sub-process task Location properties	“Setting location properties for Bravo pipette tasks” on page 94
Liquid classes	“Creating a liquid class” on page 173
Creating protocols	“Workflow for creating a protocol” on page 73
About sub-processes	“Setting Sub Process (Bravo) task parameters” on page 119

Setting Dispense (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Dispense task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Dispense task in a protocol

Dispense task defined

A Dispense task dispenses liquid into a plate.

This task is available in a Bravo sub-process.

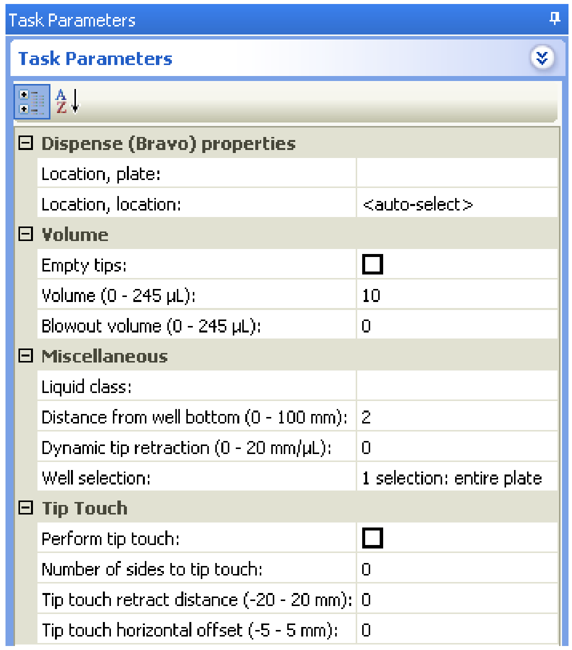
Procedure

To set the Dispense (Bravo) task parameters:

1. Add the **Dispense (Bravo)** task to a protocol sub-process.
2. Set the **Dispense (Bravo) properties**.

Property	**Description**
Location, plate	Identifies the plate at which the dispense will occur.
Location, location	Identifies the location at which the dispense will occur. If you select autoselect, then the robot will choose an available location. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.
Empty tips	Indicates whether to empty entire contents of tips, including fluid and air. Volume parameter is ignored if this option is checked.
Volume	Specifies the volume of liquid to be dispensed from each pipette tip.
Blowout volume	Specifies the volume of air to dispense after the main volume has been dispensed. Typically, the blowout volume is the same as the pre-aspirate volume. *Note:* Blowout only occurs in the last quadrant dispensed for a given Dispense task.
Liquid class	Indicates you have defined a liquid class for this liquid.
Distance from well bottom (0–100 mm)	Specifies the distance from the well bottoms that the tips will be at the start of the dispense task.

Property	Description
Dynamic tip retraction (0–20 mm/µL)	Specifies the distance that the tips will move upwards per unit volume of liquid being dispensed. For an approximation, use well volume/well depth.
Well selection	Identifies wells for dispensing. Applies only if the Bravo head has fewer tips than the plate has wells, or if you are in single row/column dispensing mode. Click the ellipsis button, and, in the Well Selection dialog box, click wells, rows, or columns to highlight the wells into which liquid is dispensed.
Perform tip touch	Specifies whether a tip touch is performed after each selection of the plate.
Number of sides to tip touch	Specifies the number of places on the side of the well that the tip touches after dispensing. The number ranges from 0 to 4.
Tip touch retract distance (-20–20 mm)	Specifies the height that the tips move up in addition to the distance from the well bottom before touching the sides of the wells.
Tip touch horizontal offset (-5–5 mm)	When the value for this parameter is 0, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.



Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119
Setting Location properties	"Setting location properties for Bravo pipette tasks" on page 94
Liquid classes	"Creating a liquid class" on page 173

Setting Mix (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Mix task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Mix task in a protocol

Mix task defined

A Mix task aspirates and dispenses liquid multiple times to mix it.

This task is available in Bravo sub-processes.

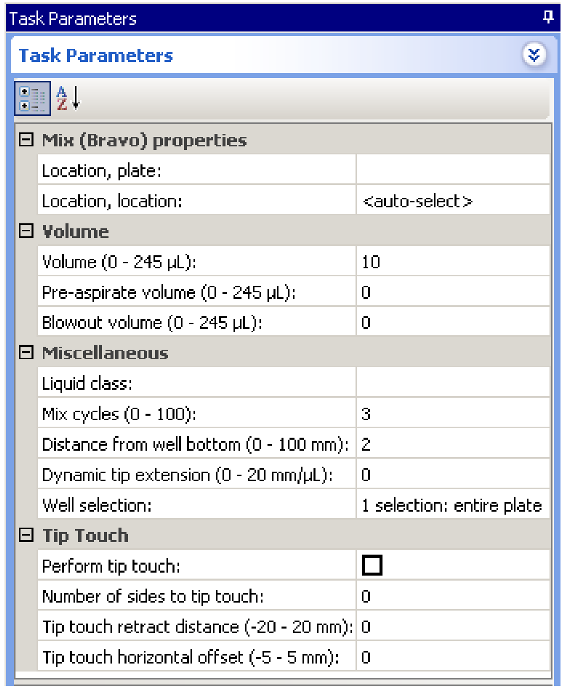
Procedure

To set the Mix (Bravo) task parameters:

1. Add the **Mix (Bravo)** task to a protocol sub-process.
2. Set the **Mix (Bravo) properties**.

Property	Description
Location, plate	Identifies the plate at which the mix will occur.
Location, location	Identifies the location at which the mix will occur. If you select autoselect, then the robot will choose an available location. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.
Volume	Specifies the volume of liquid to be mixed for each plate well.
Pre-aspirate volume	Specifies the volume of air to be drawn before the pipette tips enter the liquid.
Blowout volume	Specifies the volume of air to dispense when the tips are in the liquid of the last quadrant after the last cycle. Typically the same as the pre-aspirate volume.
Liquid class	Indicates you have defined a liquid class for this liquid.
Mix cycles (0–100)	Specifies the number of aspirate/ dispense operations.
Distance from well bottom (0–100 mm)	Specifies the starting distance from the well bottoms that the tips will be during the mix cycle.

Property	Description
Dynamic tip extension (0–20 mm/µL)	Specifies the distance that the tips will move downwards and upwards per unit volume of liquid being dispensed or aspirated. For an approximation, use well volume/well depth.
Well selection	Identifies wells for dispensing. Applies only if the Bravo head has fewer tips than the plate has wells, or if you are in single row/column mode. Click the ellipsis button, and, in the Well Selection dialog box, click wells, rows, or columns to highlight the wells into which liquid is dispensed.
Perform tip touch	Specifies whether a tip touch is performed after each selection of the plate.
Number of sides to tip touch	Specifies the number of places on the side of the wells that the tips touch after the last cycle of each quadrant. The number ranges from 0 to 4.
Tip touch retract distance (-20–20 mm)	Specifies the height that the tips move up before touching the sides of the wells.
Tip touch horizontal offset (-5–5 mm)	When the value for this parameter is 0, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.



Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119
Sub-process task Location properties	"Setting location properties for Bravo pipette tasks" on page 94
Liquid classes	"Creating a liquid class" on page 173

Setting Move to Location (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Move To Location task. Read this topic if you are:

- An administrator or technician who writes protocols
- An operator who needs to edit the Move To Location task in a protocol

Move To Location task defined

This task moves the pipette head to a safe distance above a given deck location. The safe distance is set in the Bravo profile.

Typically, this task is used in startup and cleanup protocols to move the pipette head out of the way. For example, move the pipette head to location 9 before replacing a plate on location 1.

The Move To Location (Bravo) task can be used in Bravo

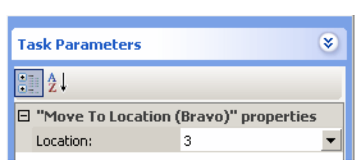
- sub-processes
- Startup protocols
- Cleanup protocols

Procedure

To set the Move To Location (Bravo) task parameters:

1. Add the **Move To Location (Bravo)** task to a protocol sub-process.
2. Set the **Move to Location (Bravo) properties**.

Property	**Description**
Location	Specifies the *x*, *y* location the pipette head will move to by identifying either its number on the deck (1–9) or the plate name. The height above the deck that the head moves to is determined by the *z*-axis safe location set in the profile.



Related topics

For information about...	**See...**
Creating protocols	"Workflow for creating a protocol" on page 73

For information about...	See...
Bravo deck locations	*Bravo User Guide*

Setting Pump Reagent (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Pump Reagent task that is used with a Velocity11 Pump Module. A Pump Module has two pumps that can be set to pump forward or in reverse.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Pump Reagent task in a protocol

Pump Reagent task defined

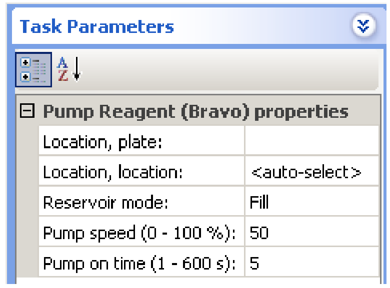
The task fills or empties the Auto Filling Reservoir and Microwash Reservoir by pumping for a specified number of seconds. If the reservoir is on a Weigh Station, the pump stops fluid flow when the target weight is reached.

Procedure

To set the Pump Reagent (Bravo) task parameters:

1. Add the **`Pump Reagent (Bravo)`** task to a protocol sub-process.
2. Set the **`Pump Reagent (Bravo) properties`**.

Property	Description
Location, plate	Identifies the plate at which the pump reagent task will occur.
Location, location	Identifies the location of the reservoir or Weigh Station. *Note:* If you are using the Configure Labware task, this entry is ignored.
Reservoir mode	Specifies: ❑ Fill if you want the reservoir to be filled ❑ Empty if you want the reservoir to be emptied In Bravo Diagnostics you define which pump performs which task.
Pump speed (0–100%)	Specifies the percentage of the maximum pump speed.
Pump on time (0–600 s)	Specifies the pump time in seconds.



Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119
Configuring the Pump Module in diagnostics	*Bravo User Guide*
Sub-process task Location properties	"Setting location properties for Bravo pipette tasks" on page 94

Setting the Reserve Location task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Reserve a Location task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Reserve Location task in a protocol

Reserve Location task defined

This task instructs VWorks to not put or place a piece of labware at the selected location. Typically, this task is used to delay a pipetting action. For example, you might want samples to incubate for a period of time before aspirating.

The Reserve Location (Built In) task is available for sub-processes.

Procedure

To set the Reserve Location task parameters:

1. Add the **Reserve Location (Built In)** task to a protocol sub-process.
2. Set the **Reserve Location (Built In) properties**.

Property	Description
Location to use	Location to reserve. Select from the list.
Reservation time	Type the time in seconds.

Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119

Setting Serial Dilution (Bravo) task parameters

About this topic

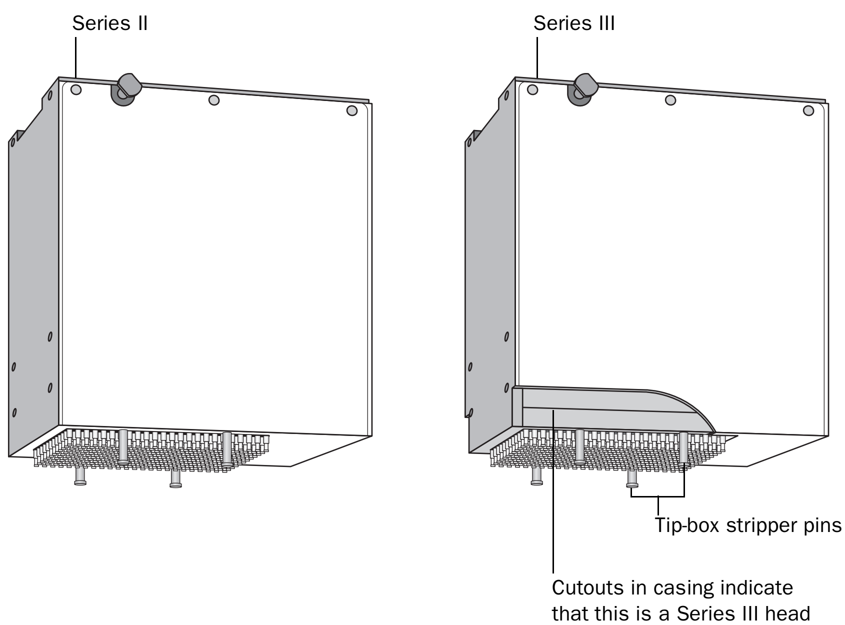
This topic describes how to set the parameters for the Serial Dilution task. (The Serial Dilution task is common to many automated pipettors in VWorks.)

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Serial Dilution task in a protocol

Row and column serial dilution requires a Series III head

If you are using Series II pipette head from a VPrep instrument, you can only use the Serial Dilution task if you have more wells than pipette tips enabling you to dilute by quadrant, not by single-row or single-column. You can recognize a Series II pipette head because it does not have cutouts in the casing.



Out-of-bounds plate locations

Typical serial dilutions use single-row or single-column pipetting. This means some locations on the deck cannot be entirely accessed by the pipette head but, they can be used for tip trash and reservoirs.

Note: Reservoirs can only be used if you select columns 12 or 24, or rows H or P.

If pipetting by...	Don't put plates or tips on locations...
Single-row	1, 2, or 3
Single-column	1, 4, or 7

Serial Dilution task defined

The Serial Dilution task allows you to set up a serial dilution in a plate using a single task. Configuring the Serial Dilution task settings is performed using a wizard that guides you through the setup process. The end result is a series of Aspirate, Dispense, and Mix (optional) tasks that produce a linear or non-linear concentration gradient in a selected plate area.

Pipette head and microplate configurations

The Serial Dilution task can be used with the following head/plate configurations:

Pipette head	Microplate well number
Series II 8-channel head	96, 384
Series II 16-channel head	384, 1536
Series II 96-channel head	384, 1536
Series II 384-channel head	1536
Series III 96-channel head with disposable tips	96, 384, 1536
Series III 384-channel head with disposable tips	384, 1536

Before you start

The Serial Dilution task can only be used with a microplate and not a reagent plate or reservoir.

Before you start, make sure the serial dilution plate has the following:

- ❑ A column, row, or quadrant that contains the starting concentration of a compound to be diluted
- ❑ One or more columns, rows, or quadrants each containing the same amount of diluent

Procedure

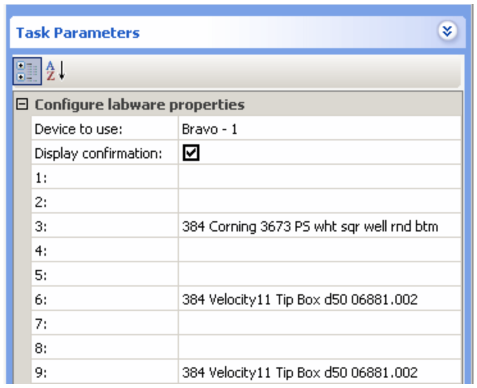
To use the Serial Dilution task:

- ❑ Configure the labware on the Bravo deck.
- ❑ Set the head mode. The head mode instructs the software whether you are performing a row, columns, or quadrant dilution.
- ❑ Add the Serial Dilution task and configure the parameters using the serial dilution wizard.

To configure the labware:

1. In the main protocol editor, drag the **Configure Labware** task icon from the **Available Tasks** list to the main protocol process.

2. Under **Task Parameters**, configure the locations. In the example shown, location 3 is configured for the dilution plate. Because in this example we want to change tips between each dilution, location 6 is configured for loading tips and location 9 for unloading tips.



To set the head mode:

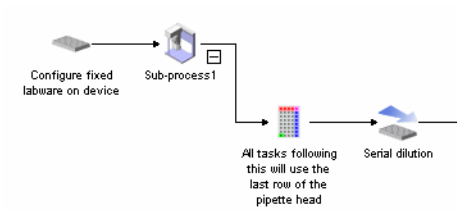
1. Because all pipetting tasks can only be used as part of a sub-routine of the main protocol, add a **SubProcess** task to the protocol.
2. Add the **Set Head Mode** task to the **SubProcess** task.
3. In the **Task Parameters**, select a **Head mode** from the list.

   If you are using a Series II head, select **Use all barrels on head**.

To set the Serial Dilution task parameters:

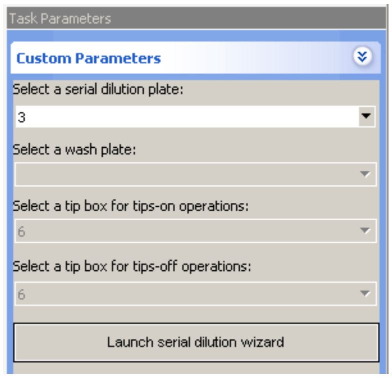
1. Add the **Serial Dilution** task after the **Set Head Mode** task.

   If you will not be changing tips during the dilution series, you will need to add a **Tips On** task and set its parameters before the **Serial Dilution** task is performed.



2. In **Task Parameters**, select the deck location of the plate you are using for the serial dilution. In the example shown, it is **3**.

   The rest of the settings for the serial dilution are made in the dilution wizard.



To use the serial dilution wizard:

1. Before you start the serial dilution wizard, be prepared to answer the following questions:
   - Are you doing a fixed volume or variable volume dilution?
   - Do you want to perform mixing, tip washing, or tip changing?
   - Will you be starting with tips on or off the head?
   - Which column, row, or quadrant has the starting material?
   - Which columns, rows, or quadrants will be diluted?
   - What settings do you want to use for the Aspirate, Mix, and Dispense tasks?
2. Click **Launch serial dilution wizard**.

   In step 1 of the wizard, choose whether you want to specify the volume and concentration parameters.

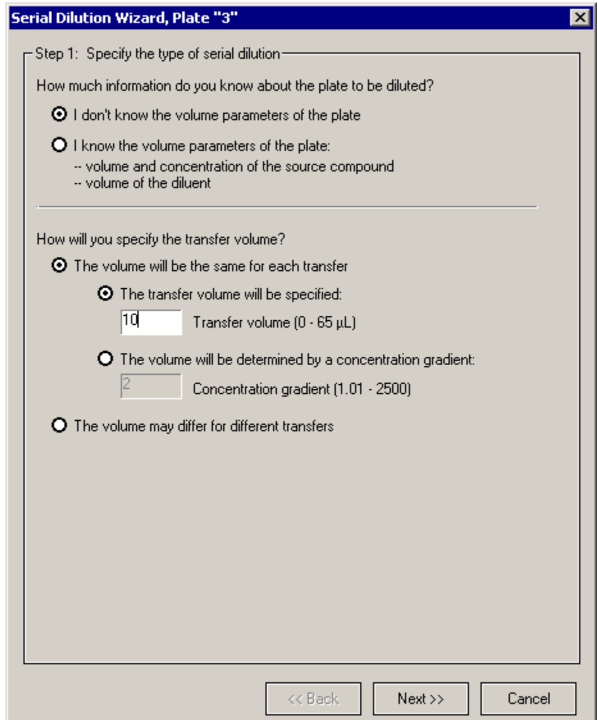
   a. Specify the information you have about the plate.

      Knowing these parameters enables you to:

      – Determine the dilution according to a desired concentration gradient. The wizard calculates the necessary transfer volumes.

      – View resultant concentration if you choose to specify volumes.

      In this example, the parameters of the plate are not known.



b. Choose how the transfer volumes will be determined.

   A fixed transfer volume results in a dilution series whose concentration gradient is linear.

   A variable transfer volume results in a dilution series whose concentration gradient is non-linear.

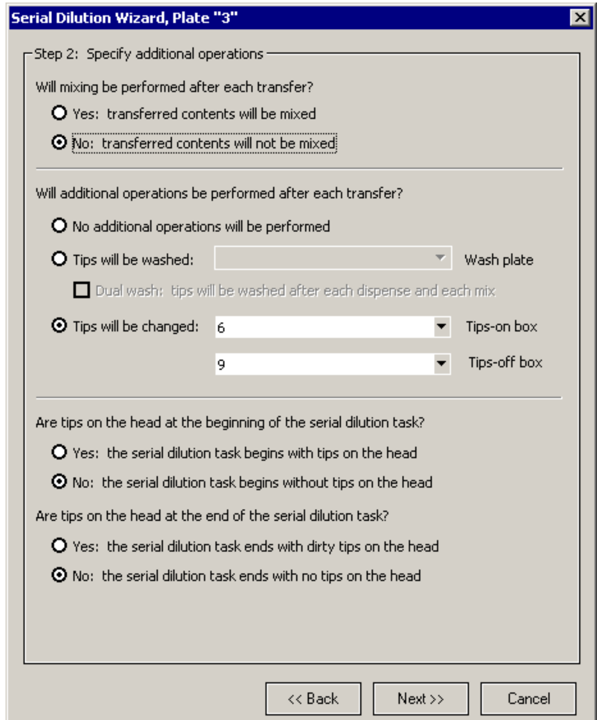
   If you choose a constant transfer volume, either enter the volume to be transferred in the **Transfer volume** field or select **The volume will be determined by a concentration gradient** and enter the dilution factor in the **Concentration gradient** field.

   For example, if the **Concentration gradient** = 2, then the concentration of the first dilution will be the concentration in the starting column (n) divided by 2, or n/2. The concentration of the second dilution will be the concentration of the second column (n–1) divided by 2, or (n–1)/2 and so forth.

   Note: The upper range of the transfer volume is determined by the capacity of the pipette head and the well volume of the plate.

   In our example, we will be transferring the same specified volume for each dilution.

3. Click **Next** to go to step 2 of the wizard, and enter information about additional operations.

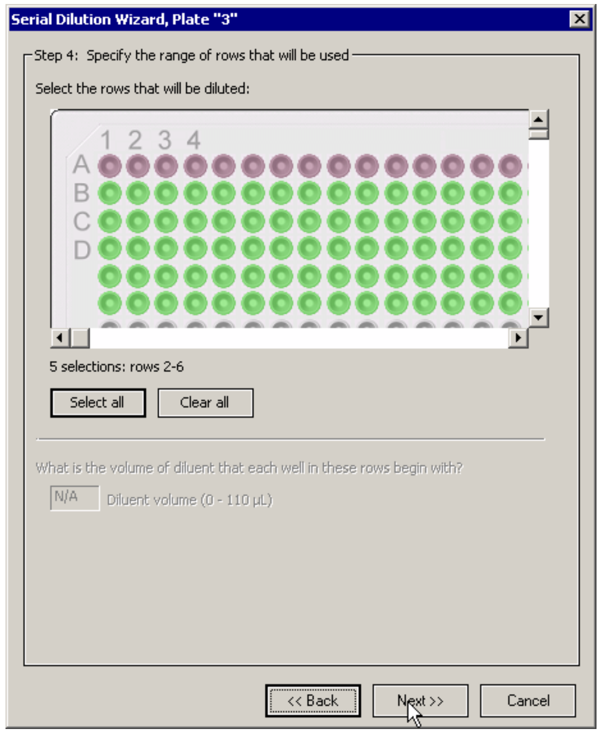


In the example shown, there is no mixing samples or washing tips but the tips will be changed between each dilution. The location to load tips is 6 and the location to get rid of tips is 9.

4. Click **Next** to go to step 3 of the wizard and select the column, row, or quadrant of the plate that contains the starting concentration.

   Note: Only one column, row, or quadrant can be selected.

5. Click **Next**, to go to step 4 of the wizard, and enter information about the dilution wells.



Specify the columns, or quadrants of wells that will be used in the dilution.

If you selected **I know the volume parameters of the plate** in step 1 of the wizard, enter the volume in each well in the **Diluent volume** field.

6. Click **Next** to go to step 5 of the wizard to specify and verify information about the transfers.

   If you selected **I know the volume parameters of the plate** in step 1 of the wizard, the final concentration for each dilution step can be viewed in the **Resultant concentration** field.

   To view the information about a particular column, row, or quadrant, click **Next transfer** or **Previous transfer**.

   If you selected **Volume may differ for different transfers** in step 1 of the wizard, examine each transfer volume by clicking **Next transfer** and adjust the values in the **Transfer volume** if necessary.

7. Click **Next** to go to step 6 of the wizard and enter the aspirate and dispense property parameters for these tasks that will take place during the serial dilution process.

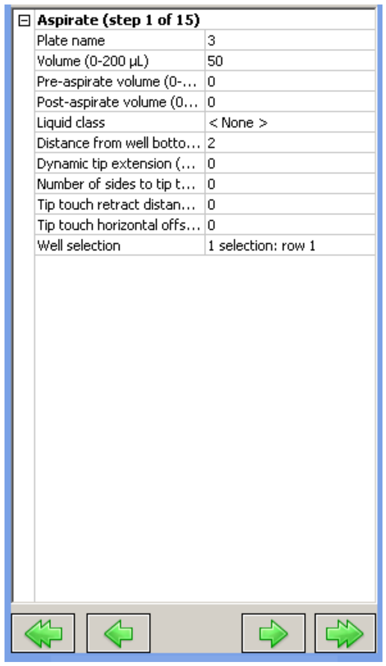
   Note: These values are applied to all **Aspirate** and **Dispense** tasks that occur within the **Serial Dilution** task. The **Volume** and **Well selection** fields are not available because this information was entered in steps 2 and 3 of the wizard.

8. Click **Next** to go to step 7 of the wizard and if you are mixing during the dilution, enter the mix property parameters.

9. Click **Finish**.

   The **Pipette Task Parameters** toolbar displays the properties of the first step in the **Serial Dilution** task.

   Arrows appear at the bottom of the toolbar. Use these to scroll through each step of the serial dilution.



Executing the Serial Dilution task

VWorks treats the Serial Dilution task as a series of tasks. Each task within the serial dilution is recorded with its own time stamp in the log file.

Related topics

For information about...	See...
Setting the head mode	"Setting Set Head Mode (Bravo) task parameters" on page 117
Retracting stripper pins before doing single-row pipetting	*Bravo User Guide*
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119

Setting Set Head Mode (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Set Head Mode task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Set Head Mode task in a protocol

Set Head Mode task defined

With the Series III pipette head, you can choose to pipette with just one row or one column of tips instead of using all tips on the head.

Pipetting with one column or row allows you to do serial dilutions.

A Set Head Mode (Bravo) task specifies whether all the barrels in the pipette head are used for pipetting or just the outer row or column are used.

This task is available for Bravo sub-processes and only if the Series III pipette head is installed.

Before you start

If you are going to use the head to pipette the last row of barrels, you will need to retract the tip box stripper pins on the pipette head. See Related topics at the end of this topic to locate this procedure.

Procedure

To set the Set Head Mode (Bravo) task parameters:

1. Add the **Set Head Mode (Bravo)** task to a protocol sub-process.
2. Set the **Set Head Mode (Bravo) properties**.

Property	Description
Head mode	Possible values are: ❑ Use all barrels on head ❑ Use the last column of barrels on head (limited use of deck locations 1, 4, or 7) ❑ Use the last row of barrels on head (limited use of deck locations 1, 2, or 3)

Related topics

For information about...	See...
Retracting the tip box stripper pins	*Bravo User Guide*
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119

Setting Shake (Bravo) task parameters

About this topic

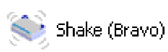
This topic describes how to set the parameters for the Bravo’s Shake task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Shake task in a protocol

Shake task defined

The Shake (Bravo) task shakes instructs the location that has an Orbital Shaking Station.

This task is available for sub-processes and only if you have an Orbital Shaking Station.



Procedure

To set the Shake task parameters:

1. Add a **Shake (Bravo)** task to a protocol sub-process.
2. Set the **Shake (Bravo) properties**.

Property	Description
Location, plate	Identifies the plate at which the shake will occur.
Location, location	Identifies the location at which the shaking will occur. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.
Mode	Specifies shake mode: ❑ Continuous (On, Off) ❑ Timed
RPM	Specifies the shake speed to shake in revolutions per minute.
Direction	Specifies the direction to shake. Choose from the list.
Time for Operation	Specifies duration to shake in Timed mode (in seconds).

Related topics

For information about...	See...
Using the Sub-Process task	“Setting Sub Process (Bravo) task parameters” on page 119
Installing an Orbital Shaking Station	*Bravo User Guide*

For information about...	See...
Sub-process task Location properties	"Setting location properties for Bravo pipette tasks" on page 94

Setting Sub Process (Bravo) task parameters

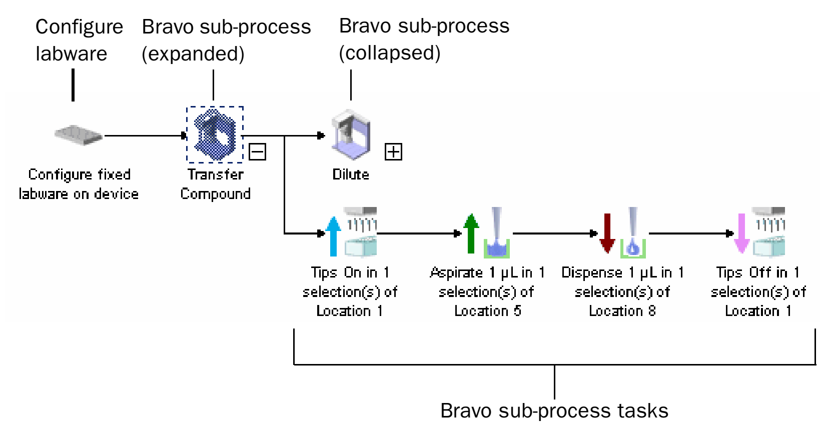
About this topic

This topic describes how to set the parameters for the Bravo's Sub Process task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Sub Process task in a protocol

Sub Process task defined

All the Bravo's pipetting tasks (Aspirate, Dispense, Mix, and so on) must belong to a Bravo sub-process. The Sub Process icon can be expanded to show all the pipetting tasks or collapsed to make longer processes easier to read.



Procedure

To set the Sub Process (Bravo) task parameters:

1. Add the **SubProcess (Bravo)** task to a protocol process.
2. Set the **SubProcess (Bravo) properties**.

Property	Description
Sub-process name	Enter a name for the sub process.

Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73

For information about...	See...
Protocols	"About tasks, processes, and protocols" on page 68

Setting Tips On (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Tips On task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Tips On task in a protocol

Tips On task defined

A Tips On (Bravo) task puts fresh tips on the pipette head.

This task is available in Bravo sub-processes and only with a pipette head that uses disposable tips (not fixed tips).

Procedure

To set the Tips On (Bravo) task parameters:

1. Add the **Tips On (Bravo)** task to a protocol sub-process.
2. Set the **Tips On (Bravo) properties**.

Property	Description
Location, plate	Identifies the plate at which the tips will be installed.
Location, location	Identifies the location at which the tips will be installed. If you select autoselect, then the robot will choose an available location. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.
Well selection	Specifies the tips to be picked up. Applies only if the Bravo head has fewer tips than the plate has wells or if you are in serial dilution mode. Click the ellipsis button, and, in the Well Selection dialog box, click wells, rows, or columns to highlight which tips in the tip box will be picked up.

Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119
Sub-process task Location properties	"Setting location properties for Bravo pipette tasks" on page 94

Setting Tips Off (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Tips Off task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Tips Off task in a protocol

Tips Off task defined

A Tips Off (Bravo) task removes the pipette tips from the pipette head.

This task is available in Bravo sub-processes and only with a pipette head that uses disposable tips (not fixed tips).

Procedure

To set the Tips Off (Bravo) task parameters:

1. Add the **`Tips Off (Bravo)`** task to a protocol sub-process.
2. Set the **`Tips Off (Bravo) properties`**.

Property	Description
Location, plate	Identifies the plate at which the tips will be removed.
Location, location	Identifies the location at which the tips will be removed. If you select autoselect, then the robot will choose an available location. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.

Property	Description
Well selection	Specifies the tips to be removed. Applies only if the Bravo head has fewer tips than the plate has wells or if you are in serial dilution mode. Click the ellipsis button, and, in the Well Selection dialog box, click wells, rows, or columns to highlight where in the tip box or tip trash the removed tips will be placed.

Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
About sub-processes	"Setting Sub Process (Bravo) task parameters" on page 119
Sub-process task Location properties	"Setting location properties for Bravo pipette tasks" on page 94

Setting User Message task parameters

About this topic

This topic describes how to set the parameters for the User Message task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the User Message task in a protocol

User Message task defined

The administrator or technician who creates a protocol can add User Message tasks to provide reminders to the operator. The reminders are in the form of messages that appear on the screen at the appropriate time and pause the protocol until acknowledged by the operator.

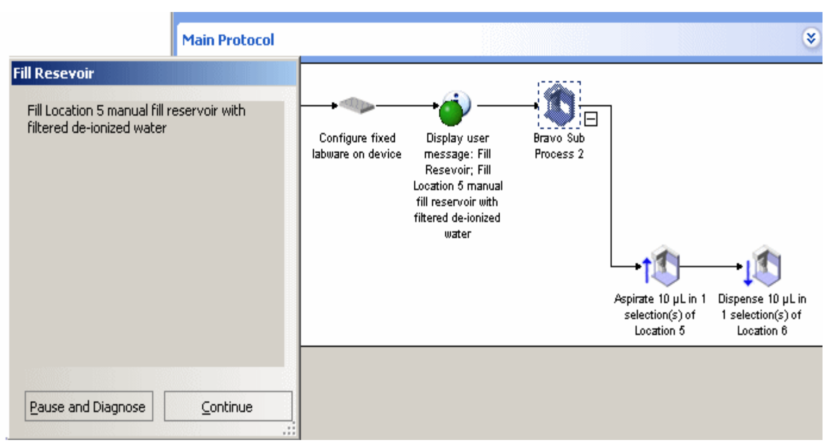
User message tasks can, for example, be used to remind the operator to empty the waste container, fill a reservoir, or remove plates.

Note: User messages do not appear when running a protocol in simulation mode.

!! IMPORTANT !! Remember to remove all user messages from protocols that you intend to run unattended.

Example

In the example shown below, the protocol has a user message that reminds the operator to fill a reservoir.



Procedure

To set User Message task parameters:

1. Add the **User Message (BuiltIn)** task to a protocol process.
2. Set the **User Message properties**.

Property	Description
Title	Name of the user message, such as fill reservoir.

Property	Description
Body	Details about the task, such as which locations to fill.
Only show the first time	Option to execute the task during the first protocol cycle of a repeating protocol run.

Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73

Setting Vacuum filter task parameters

About this topic

This topic describes how to set the parameters for the Vacuum filter task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Vacuum Filter task in a protocol

Vacuum Filter task defined

The Vacuum filter (Bravo) task instructs the location that has a vacuum filter to apply a vacuum.

This task is available in Bravo sub-processes and only if you have a Vacuum Filtration Station installed on your Bravo.

Procedure

To set the Vacuum filter task parameters:

1. Add the **Vacuum filter (Bravo)** task to a protocol sub-process.
2. Set the **Vacuum filter (Bravo) properties**.

Property	Description
Location, plate	Identifies the plate at which the vacuum will be applied.
Location, location	Identifies the location at which the vacuum will be applied. *Note:* If you are using the Configure Labware task for this location, this entry is ignored.
Mode	Specifies the possible operation modes: ❑ Continuous (On, Off) ❑ Timed

Property	Description
Time for Operation	Specifies the vacuum duration in seconds.

Related topics

For information about...	See...
Installing a Vacuum Filtration Station	*Bravo User Guide*
Adding a sub-process	"Setting Sub Process (Bravo) task parameters" on page 119
Creating protocols	"Workflow for creating a protocol" on page 73
Sub-process task Location properties	"Setting location properties for Bravo pipette tasks" on page 94

Setting Wait For User (Bravo) task parameters

About this topic

This topic describes how to set the parameters for the Bravo's Wait For User task. Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to edit the Wait For User task in a protocol

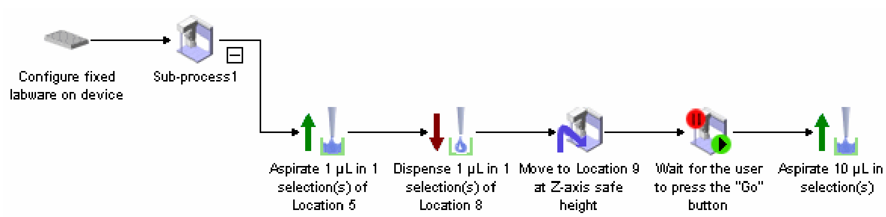
Wait For User task defined

A Wait For User (Bravo) task pauses the protocol and waits until the operator has pressed the go button on the pendant. When the button is pressed, the protocol continues on to the next task.

Usage example

For example, you are using the Bravo (without the gripper arm) in a laminar flow hood. When dispensing is finished on a plate, you want the protocol to pause while you remove the plate and replace it with a fresh plate. Use the Wait For User (Bravo) task to pause the protocol. Then, when the new plate is in its location, press the go button to continue the protocol.

You might also want to put a Move To Location task before the Wait For User task to move the pipette head out of the way. See the following example.



Note: The User Message task is another way to insert a pause in a protocol. If you have easy access to the computer during a run, the User Message task is a better way to add a pause to the run.

Procedure

There are no task parameters to set for this task. When you are ready to continue with the run, press the go button.

Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73
Bravo tasks	"List of Bravo task parameters" on page 90

Setting Wash Tips (Bravo) task parameters

About this topic

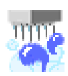
This topic describes how to set the parameters for the Bravo's Wash Tips task. Read this topic if:

- Your Bravo is equipped with a Pump Module and a MicroWash Reservoir
- You are:
  - An administrator or technician who writes protocols
  - An operator who needs to edit the Wash Tips task in a protocol

Wash Tips defined

A Wash Tips task is used with a Bravo to wash pipette tips and prime the Microwash Reservoir manifolds.

This task is available in Bravo sub-processes and only if a Microwash Reservoir is installed.


Wash Tips (Bravo)

Procedure

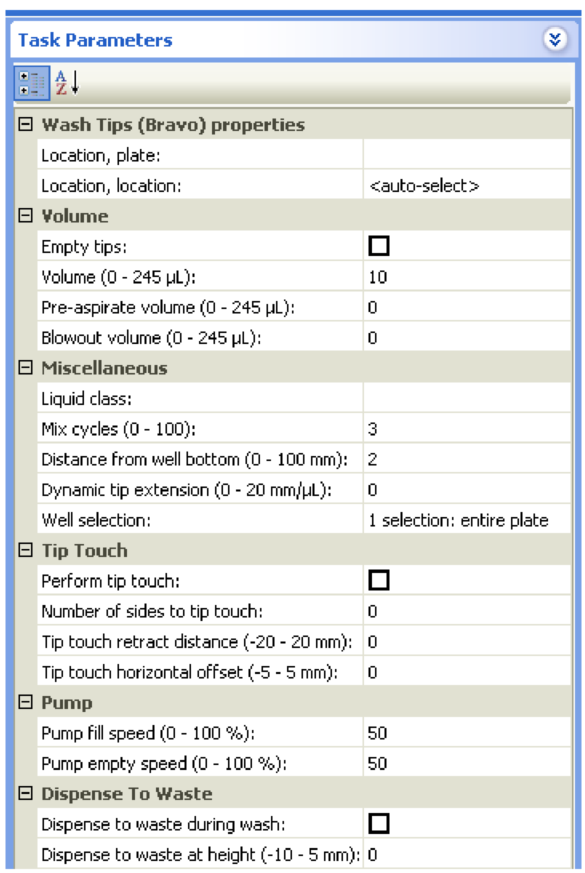
To set the Wash Tips (Bravo) task parameters:

1. Add the **Wash Tips (Bravo)** task to a protocol sub-process.
2. Set the **Wash Tips properties**.

Property	**Description**
Location, plate	Identifies the plate at which the tip washing will occur.
Location, location	Identifies the location at which the tip washing will occur. If you select autoselect, then the robot will choose an available location. *Note:* If you are using the Configure Labware task for this location, this entry is ignored. We recommend that you use locations 1, 2, or 3 to keep tubing away from the Bravo deck.
Empty tips	Specifies emptying entire contents of tips, including fluid and air. Volume parameter is ignored if this option is checked.
Volume	Specifies the volume of liquid to be aspirated and dispensed to each plate well.
Pre-aspirate volume	Specifies the volume of air to be drawn before the pipette tips enter the liquid.

Property	Description
Blowout volume	Specifies the volume of air to dispense when the tips are in the liquid of the last quadrant after the last cycle. Typically the same as the pre-aspirate volume.
Liquid class	Indicates if you have defined a liquid class for this liquid.
Mix cycles (0–100)	Specifies the number of aspirate/ dispense operations.
Distance from well bottom (0–100 mm)	Specifies the minimum distance from the bottoms of the plate wells or MicroWash chimneys that the tips will be during a wash cycle.
Dynamic tip extension (0–20 mm/µL)	Specifies the distance that the tips should move upwards or downwards per unit volume of liquid being dispensed or aspirated. You need to determine an appropriate value by trial-and-error.
Well selection	Specifies the tips to be washed. Applies only if the Bravo head has fewer tips than the plate has wells. Click the ellipsis button, and, in the Well Selection dialog box, click wells to highlight which MicroWash chimneys the tips will be washed in.
Perform tip touch	Specifies whether a tip touch is performed after each selection of the plate.
Number of sides to tip touch	Specifies the number of places on the side of the well that the tip touches after washing. The number ranges from 0 to 4.
Tip touch retract distance (-20–20 mm)	Specifies the height that the tips move up before touching the sides of the wells.
Tip touch horizontal offset (-5–5 mm)	Specifies the horizontal distance the tips will move. When the value for this parameter is 0, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.

Property	Description
Pump fill speed (0–100 %)	Specifies the relative rate of liquid flow into the MicroWash manifold. This value should be high enough for the washing liquid to just bubble over the tops of the chimneys.
Pump empty speed (0–100 %)	Specifies the relative rate of liquid flow out of the MicroWash manifold. This value should be slightly higher than that of the inflow pump to prevent an overflow.
Dispense to waste during wash	Specifies the dispense step of the wash cycle will take place outside of the MicroWash chimneys. Dispensing to waste provides a more efficient wash than dispensing the waste into the chimneys. However, dispensing to waste takes longer because the pipette head must move more.
Dispense to waste at height (-10–5 mm)	Specifies the height at which the dispense takes place. For example, if -10 mm, the tip dispenses 10 mm below the top of the chimneys.



Related topics

For information about...	See...
Creating protocols	“Workflow for creating a protocol” on page 73
MicroWash Reservoir	*Bravo User Guide*
Pump Module	*Bravo User Guide*
About sub-processes	“Setting Sub Process (Bravo) task parameters” on page 119
Filling and emptying reservoirs	“Setting Pump Reagent (Bravo) task parameters” on page 106
Sub-process task Location properties	“Setting location properties for Bravo pipette tasks” on page 94

Setting Device Selection in task parameters

About this topic

If you are running multiple Bravos or additional devices from one instance of VWorks, you might need to specify which device you are using.

You can select a device for these two tasks:

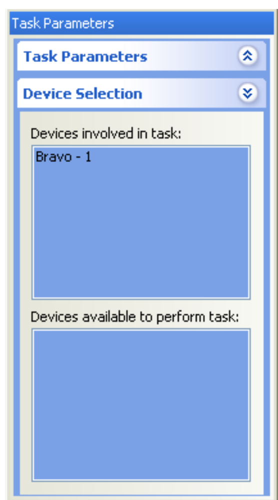
- Sub Process (Bravo) task
- Move To Location (Bravo) task in a startup or cleanup process

If you are using one Bravo in stand-alone mode, you can ignore the Device Selection portion of the Task Parameters toolbar.

Procedure

To set the devices to be used for the task:

1. In the **Task Parameters** toolbar, click the **Device Selection** tab.
2. Double-click a location or a device name to move it to or from the **Devices involved in task** list.



Related topics

For information about...	See...
Creating protocols	"Workflow for creating a protocol" on page 73

Setting Advanced Settings task parameters

About this topic

This topic shows where to set the estimated time for a task and where to enter a script.

About Estimated Time

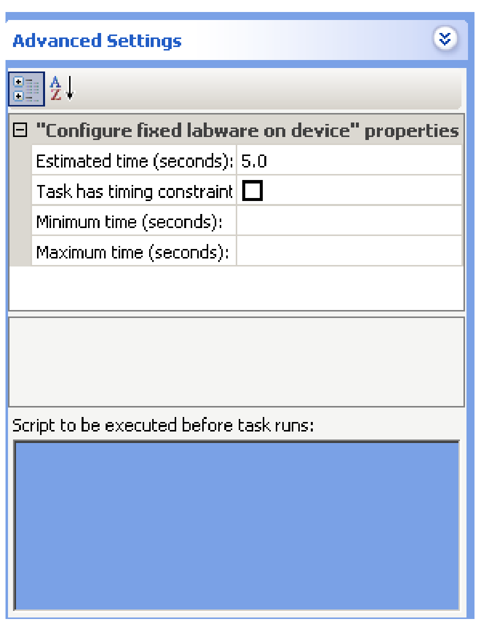
If you want a more accurate time estimate for the protocol when you run it in simulation mode, measure the time it takes for the Bravo to perform the task and then use this as Estimated Time parameter for the task.

To set the estimated time for a task

To set the estimated time for a task:

1. Select the task icon in the protocol.
2. Click the **Advanced Settings** tab.
3. Set the properties.

Property	Description
Estimated time	The estimated time to perform a task. When the simulator performs a virtual run, it uses this estimated time. Use the default supplied or adjust the times to more accurately simulate the run's timing.
Task has timing constraint	Not applicable to Bravo.
Maximum time	Minimum time the task needs to run.
Minimum time	Maximum time the task needs to run
Script to be executed before task runs:	The JavaScript you want to run before the task is performed. Enter JavaScript in the text box.



Related topics

For information about...	See...
Creating protocols	“Workflow for creating a protocol” on page 133
Writing JavaScripts	❑ “Using JavaScript in VWorks” on page 134 ❑ “JavaScripts task object and properties” on page 136

Using JavaScript in VWorks

About this topic

This topic explains how JavaScripts can be used in VWorks.

About JavaScript in VWorks

In VWorks, JavaScript can be used to:

- Configure tasks in ways that task parameters do not allow
- Change the parameters of a protocol task immediately before it is scheduled

JavaScript extends the capability of VWorks because the parameters can be changed dynamically during a run, based on:

- Information passed from an external source, such as a database
- The number of times the protocol has cycled
- Feedback on changing conditions during the run

Scripts can be run as part of startup, main, and cleanup protocols.

About JavaScript

JavaScript is a general-purpose programming language that requires an interpreter to run its programs.

You are probably most familiar with JavaScript where it is used to create dynamic effects in web pages. The form of JavaScript is made up of a core language plus web browser specific language. It is processed by the JavaScript interpreter that is built into modern browsers.

The core JavaScript language can be used to write scripts that have nothing to do with web pages. These scripts can be used for any application that includes a JavaScript interpreter. VWorks is an example of such an application—it uses a JavaScript 1.5 interpreter.

JavaScript resources

There are many JavaScript resources available online and in print. If you want to learn more about JavaScript for use in VWorks, look for resources that cover the core JavaScript language separately from the browser, client-side language and Document Object Model.

Web reference

http://www.mozilla.org.js/

Print reference

JavaScript: The Definitive Guide, Fourth Edition, published by O'Reilly.

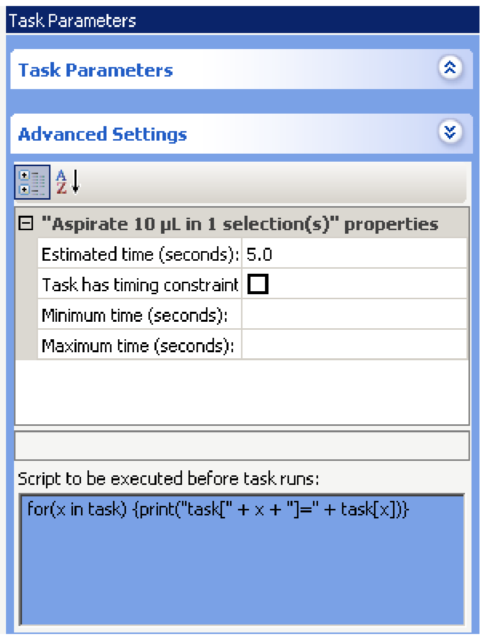
Examples of use

You can use JavaScript to:

- Print the parameters of task to the VWorks log
- Run a command line that launches an external application, such as a batch file or database updating program
- Simplify protocol writing, for example, by incrementing pipetting volumes each cycle of a protocol to perform a dilution series

Where scripts are written

Scripts are written in the **Advanced Settings** tab of the Task Parameters toolbar.



Scripts can be written two ways:

- Directly into the text box
- As an external file that is called by embedding the "`open ()`" function in the text box

How scripts are executed in protocols

Scripts are executed as:

- Separate tasks, added as a JavaScript Task to a protocol process.

  This task can be used in any protocol process.

- Links with another task, to be executed before the task is executed.

Related topics

For information about...	See...
The properties available for Bravo tasks	"JavaScripts task object and properties" on page 136
JavaScript example	"JavaScripts task object and properties" on page 136

JavaScripts task object and properties

About this topic

The VWorks JavaScript interpreter includes a task object that is defined by Velocity11.

This topic describes some common task properties and how to find the names of properties that are unique for the Bravo tasks.

Common task properties

The following properties can be used for any task.

Property	Data type	Description
task.name	String	Name of the task, for example, "Aspirate"
task.description	String	Description of the task that is given under the icon in the protocol editor. For example, if a task that has the script `print(task.description)`, VWorks writes the task description text to the log just before the task is executed.

Each task has a set of task properties

The properties available for a task correspond to its task parameters in the Task Parameters toolbar.

The name of the property is made up of "task." plus the name of the task parameter (with spaces removed). The task name is similar but not identical to the what appears in the task parameters window.

For example, the Aspirate (Bravo) task has the following task properties:

Property	Corresponds to this task parameter
task.Locationlocation	Location, location
task.Locationplate	Location, plate
task.Volume	Volume
task.Preaspiratevolume	Pre-aspirate volume
task.Postaspiratevolume	Post-aspirate volume
task.Liquidclass	Liquid class
task.Distancefromwellbottom	Distance from well bottom (0–20 mm)
task.Dynamictipextension	Dynamic tip extension (0–20 mm/µL)
task.Wellselection	Well selection
task.Performtiptouch	Perform tip touch
task.Numberofsidestotiptouch	Number of sides to tip touch

Property	Corresponds to this task parameter
task.Tiptouchretractdistance	Tip touch retract distance (0–20 mm)
task.Tiptouchhorizontaloffset	Tip touch horizontal offset (-5–5 mm)

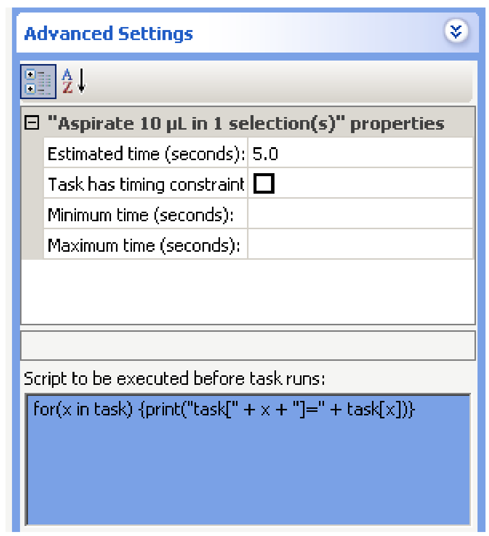
JavaScript example: Listing available properties for a task

You can generate the list of JavaScript properties available for a task by using a script.

To list the properties for a task in the Log toolbar:

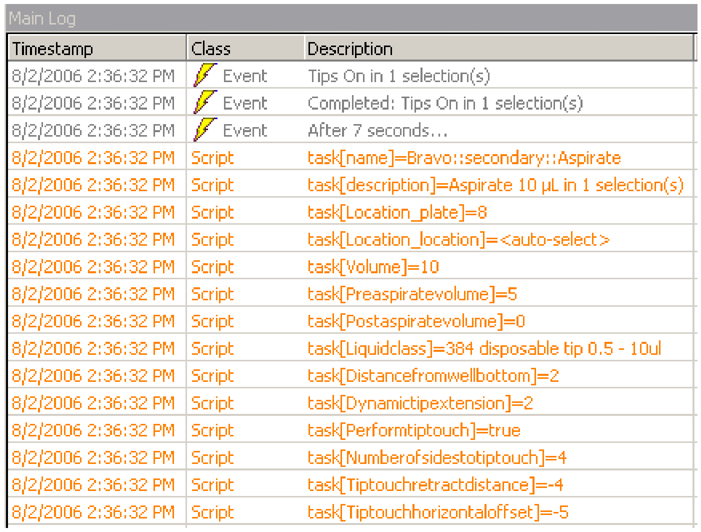
1. Create a protocol that contains the task.
2. In the **Advanced Settings** page, type this script:

```
for(x in task) {
print("task[" + x + "]=" + task[x]);
}
```



3. Run the protocol.

   The task properties available for the protocol are written to the log.



Main Log

Timestamp	Class	Description
8/2/2006 2:36:32 PM	Event	Tips On in 1 selection(s)
8/2/2006 2:36:32 PM	Event	Completed: Tips On in 1 selection(s)
8/2/2006 2:36:32 PM	Event	After 7 seconds...
8/2/2006 2:36:32 PM	Script	task[name]=Bravo::secondary::Aspirate
8/2/2006 2:36:32 PM	Script	task[description]=Aspirate 10 µL in 1 selection(s)
8/2/2006 2:36:32 PM	Script	task[Location_plate]=8
8/2/2006 2:36:32 PM	Script	task[Location_location]=<auto-select>
8/2/2006 2:36:32 PM	Script	task[Volume]=10
8/2/2006 2:36:32 PM	Script	task[Preaspiratevolume]=5
8/2/2006 2:36:32 PM	Script	task[Postaspiratevolume]=0
8/2/2006 2:36:32 PM	Script	task[Liquidclass]=384 disposable tip 0.5 - 10ul
8/2/2006 2:36:32 PM	Script	task[Distancefromwellbottom]=2
8/2/2006 2:36:32 PM	Script	task[Dynamictipextension]=2
8/2/2006 2:36:32 PM	Script	task[Performtiptouch]=true
8/2/2006 2:36:32 PM	Script	task[Numberofsidestotiptouch]=4
8/2/2006 2:36:32 PM	Script	task[Tiptouchretractdistance]=-4
8/2/2006 2:36:32 PM	Script	task[Tiptouchhorizontaloffset]=-5

Related topics

For information about...	See...
Using JavaScript	"Using JavaScript in VWorks" on page 134
Creating a protocol	"Workflow for creating a protocol" on page 73

8 Setting labware definitions

The procedures in this chapter can be performed by people with administrator or technician privileges.

This chapter contains the following topics:

- "About defining labware with the labware editor" on page 140
- "Labware editor overview" on page 142
- "Opening the labware editor" on page 145
- "Adding a labware entry" on page 147
- "Deleting a labware entry" on page 149
- "Renaming a labware entry" on page 150
- "Copying a labware entry" on page 151
- "Defining general properties" on page 152
- "Defining plate properties" on page 153
- "Defining BenchCel properties" on page 156
- "Defining stacker properties" on page 158
- "Defining pipette/well properties" on page 160
- "Defining Bravo properties" on page 162
- "Inserting an image" on page 163
- "Defining labware classes" on page 164

About defining labware with the labware editor

About this topic

This topic introduces the labware editor, which is used to define labware (plates, lids, tip boxes, and so on) that can be used by devices.

Labware editor defined

The labware editor is the VWorks interface through which you can enter information about labware.

You must be logged in as an administrator or technician to use the labware editor.

Types of information stored

Two main types of information are stored in the labware database:

- ❑ Information about the labware properties
- ❑ Information about labware classes

About labware properties

Labware has physical properties such as width, length, and number of wells. Labware can also have non-physical properties, such as robot-handling speed, offsets, and plate-handling options.

With the labware editor, all you have to do is select the type of labware to use each time you set up a protocol.

About labware classes

Labware classes are sets of labware entries, grouped so they are easier to manage than many individual labware entries.

Labware classes are used in combination with the device manager to restrict which types of labware can be used on which devices during a protocol run. This prevents wasted runs and damage to the devices on the platform.

An example of how damage can be prevented by labware restriction is where a tipbox that is too tall for a device crashes into the device as the robot delivers it.

Related topics

For information about...	See...
Labware editor	"Labware editor overview" on page 142
Opening the labware editor	"Opening the labware editor" on page 145
Defining labware classes	"Defining labware classes" on page 164

For information about...	See...
Defining properties	❑ "Defining general properties" on page 152 ❑ "Defining plate properties" on page 153 ❑ "Defining stacker properties" on page 158 ❑ "Defining pipette/well properties" on page 160

Labware editor overview

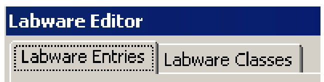
About this topic

This topic gives an overview of the organization of the labware editor's user interface.

Labware Editor pages

The labware editor has two tabs:

- *Labware Entries*. The tab contains labware definitions
- *Labware Classes*. The tab contains defined labware and class assignment



Labware Entries page

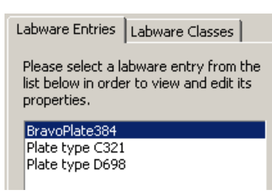
Sub-pages

The Labware Entries tab contains the following sub-tabs:

- Plate Properties
- BenchCel
- Stacker
- Pipette/Well Definition
- Bravo
- Image
- Labware Classes

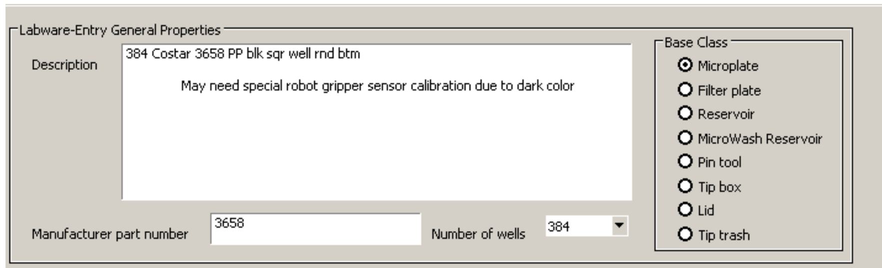
Labware selection box

The labware selection box, which is the left-hand column, is used to select a labware entry that you want to edit.



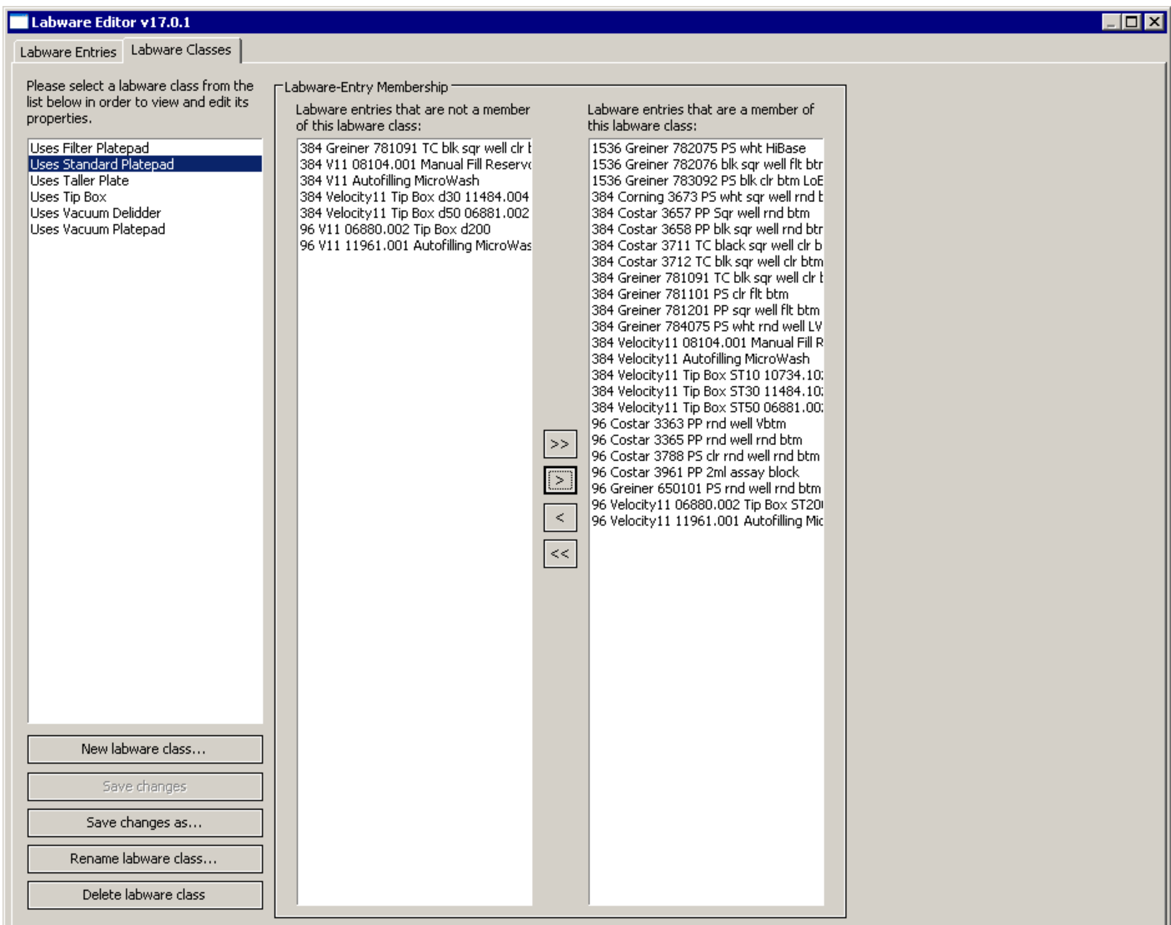
Labware-Entry General Properties area

The Labware-Entry General Properties area displays the labware-entry general properties whose selections apply across all sub-tabs.



Labware Classes page

In the Labware Classes page, you create labware classes and assign defined labware to a labware class.



Related topics

For information about...	See...
Opening the labware editor	“Opening the labware editor” on page 145
Defining labware classes	“Defining labware classes” on page 164

For information about...	See...
Defining properties	❑ “Defining general properties” on page 152 ❑ “Defining plate properties” on page 153 ❑ “Defining stacker properties” on page 158 ❑ “Defining pipette/well properties” on page 160

Opening the labware editor

About this topic

This topic explains how to open the labware editor.

You open the labware editor when you want to:

- ❑ View existing labware entries or classes
- ❑ Edit labware entries or classes
- ❑ Add new labware entries or classes
- ❑ Delete labware entries or classes
- ❑ Rename labware entries or classes

Before you start

You must be logged in as an administrator or technician to open the labware editor.

If you are adding labware, make sure you have the following:

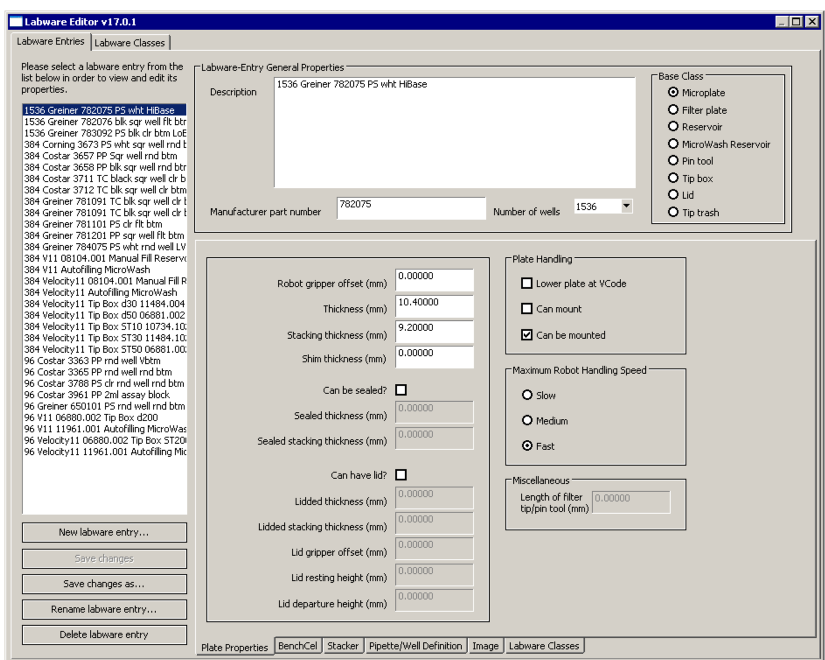
- ❑ Calipers
- ❑ Two samples of the labware you are adding

Procedure

To open the labware editor:

1. Select **Tools > Labware Editor**.

   The **Labware Editor** dialog box opens.



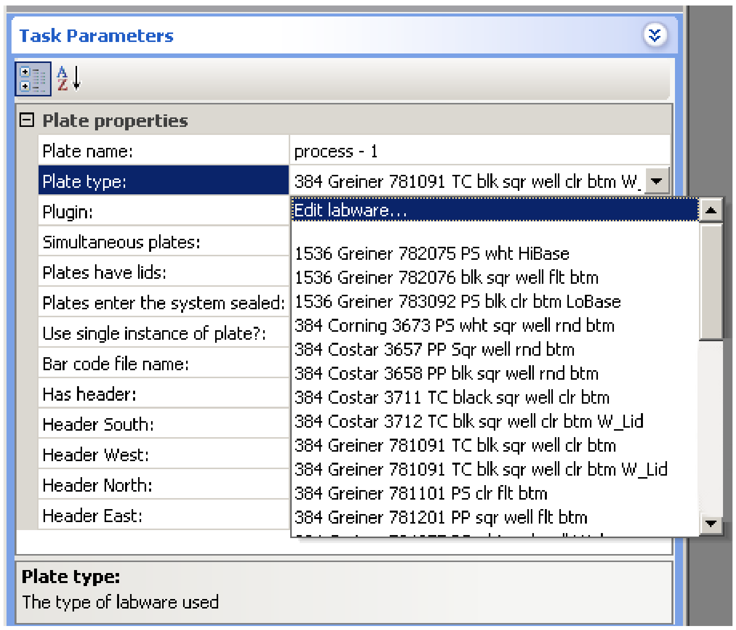
Alternate procedure

To open the labware editor from the protocol editor:

1. Click the **Protocol Editor** tab.

2. Select a plate process icon.

3. In the **Protocol Task Parameters** toolbar, select **Edit labware settings** from the **Plate type** list.



Related topics

For information about...	See...
Labware editor	"Labware editor overview" on page 142
Defining labware classes	"Defining labware classes" on page 164
Defining properties	❑ "Defining general properties" on page 152 ❑ "Defining plate properties" on page 153 ❑ "Defining stacker properties" on page 158 ❑ "Defining pipette/well properties" on page 160

Adding a labware entry

About this topic

This topic describes how to add a labware entry. The first step in defining labware, such as a plate, is to create a labware entry for it.

About this topic

The first step in defining a new piece of labware is to add a labware entry for it.

You must be logged in as an administrator or technician to perform this procedure.

Before you start

Before you add a new labware entry:

- ❑ Check to see if it is already defined in the labware editor.

  Some common labware and some Velocity11 labware comes already defined in VWorks.

- ❑ Contact Velocity11 with the definition you need.

  Velocity11 maintains a large collection of labware definitions and might be able to supply you with what you need.

Procedure

To add a labware entry:

1. Open the labware editor.
2. Under the labware selection box on the left side of the window, click **New labware entry**.

3. In the **New Labware Entry** dialog box, enter a name for the plate and click **OK**.

   For clarity, enter a detailed name for the labware that includes the manufacturer's name and plate-specific information.

   The entry appears in the labware selection box.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145

Deleting a labware entry

About this topic

If there is a labware entry that you no longer need, you can delete it. This topic describes how to delete a labware entry.

You must be logged in with an administrator or technician user account to perform this procedure.

Before you start

Make sure the entry you are deleting is not referenced in protocols.

!! IMPORTANT !! If you delete a labware entry that is already referenced in protocols, the link between the protocol and the labware data will be broken and the protocol will not run.

Procedure

To delete a labware entry:

1. Open the labware editor.
2. In the labware selection box on the left side of the window, select the labware entry to be deleted.
3. Click **Delete labware entry**.
4. In the **V11Labware** dialog box, click **Yes** to delete the entry.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145

Renaming a labware entry

About this topic

This topic describes how to change the name of a labware definition.

About this topic

You can change the name of a labware entry. In general, this is something you might do if you just named a labware type and decided to give it a different name.

You must be logged in as an administrator or technician to perform this procedure.

Before you start

Make sure either:

- ❑ The entry you are renaming is not already referenced in protocols, or
- ❑ If the entry is referenced in protocols, you update those protocols

!! IMPORTANT !! If you rename a labware entry that is already referenced in protocols, the link between the protocol and the labware data will be broken and the protocol will not run until the protocols are updated.

Procedure

To rename a labware entry:

1. Open the labware editor.
2. In the labware selection box on the left side of the window, select the labware entry to be renamed.
3. Click **Rename labware entry**.
4. In the **V11Labware** dialog box, click **Yes** to confirm that you want to rename this entry.
5. In the **Rename Labware Entry** dialog box, enter the new name for the plate and click **OK**.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145

Copying a labware entry

About this topic

To save time when creating a new entry that is similar to an existing one, you can copy an existing labware entry. This topic describes how to copy an existing labware entry.

You must be logged in as an administrator or technician to perform this procedure.

Procedure

To copy a labware entry:

1. Open the labware editor.
2. In the labware selection box on the left side of the window, select a labware entry.
3. Click **Save changes as**.
4. In the **Save Labware Entry As** dialog box, type a name for the new entry that is different from the selected one, and click **OK**.

   The copied entry appears in the labware selection box.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145

Defining general properties

About this topic

After adding a labware entry, define the general properties of the labware. This topic describes how to define the labware's general properties.

You must be logged in as an administrator or technician to perform this procedure.

About general properties

The general properties describe the type of labware that is being entered into the database and are visible on all of the sub-tabs of the labware editor.

Before you start

You must have added a labware entry that you want to define.

Procedure

To define the general properties of a piece of labware:

1. Open the labware editor.
2. Select the labware in the left column.
3. In the **Description** text box, type in a description of the labware and any other useful information.
4. For your reference, in the **Manufacturer part number** text box, enter the appropriate number.
5. In the **Number of wells** list, select the number of wells in the plate.

   If you are defining a tip box, this is the number of tips that the box can hold.
6. In the **Base Class** area, select one of the options.

   The option you select determines which labware editor properties are available. For example, when a base class of **Microplate** is selected, the **Length of filter tip/pin tool (mm)** property is unavailable.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145
Creating a labware entry (before defining it)	"Adding a labware entry" on page 147
What to do next	"Defining plate properties" on page 153

Defining plate properties

About this topic

This topic describes the parameters used to define a plate or other piece of labware in the Plate Properties sub-tab of the labware editor.

You must have a technician or administrator user account to perform this procedure.

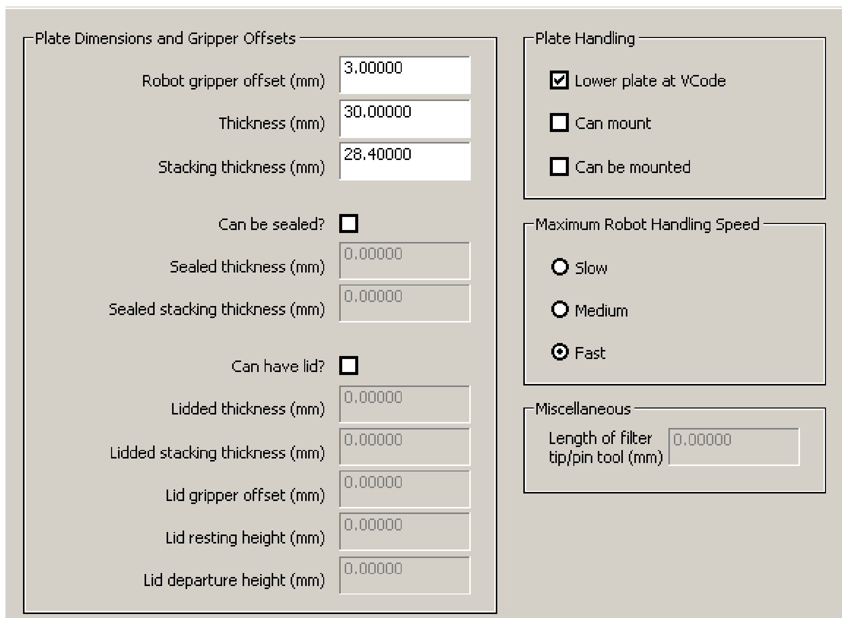
Defining plate properties

To define plate properties:

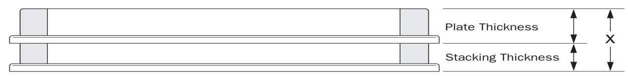
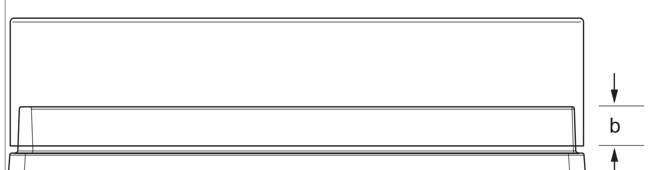
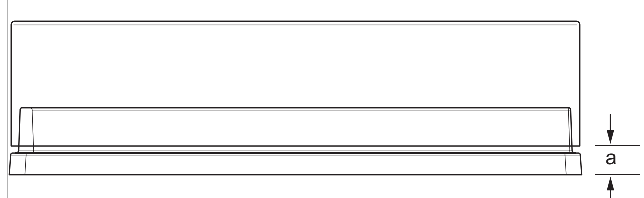
1. Click the **Plate Properties** sub-tab of the labware editor.
2. Enter the values for the available parameters according to the type of plate or labware you are defining.

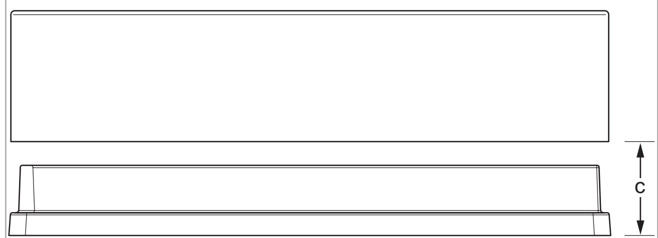
   The parameters on the **Plate Properties** sub-tab are described in the following screenshot and table.

   Note: Only the parameters associated with the **Base Class** you selected in the **General Properties** section will be available.



Property	Description
Robot gripper offset	Height of the gripper above any teachpoint when the robot is picking or placing a plate of this type. The value is typically 0–3 mm.
Thickness	The distance, in millimeters, from the bottom surface of the plate to the top surface of the plate. For a tip box, this is the distance from the bottom surface of the box to the top of the tips. Measure the distance using calipers.

Property	Description
Stacking thickness	The thickness, in millimeters, of two stacked plates minus the thickness of one plate. Measure the distance using calipers. Example: Thickness of two stacked plates (*x*) = 23.14 mm Thickness of one plate = 14.14 mm Stacking thickness: 23.14 mm - 14.14 mm = 9.00 mm 
Can be sealed?	The option to include the plate seal.
Sealed thickness	The thickness, in millimeters, of the plate with a seal in place. Available only if Can be sealed? is selected.
Sealed stacking thickness	The stacking thickness, in millimeters, of the plate with a seal in place. Available only if Can be sealed? is selected.
Can have lid?	The option to include a plate lid.
Lidded thickness	The thickness, in millimeters, of the plate with a lid in place. Available only if Can have lid? is selected.
Lidded stacking thickness	The stacking thickness, in millimeters, of the plate with the lid in place. Available only if Can have lid? is selected.
Lid gripper offset	The height, in millimeters, above the lid resting height at which to grip the lid. (Shown as *b* below.) 
Lid resting height	The height, in millimeters, above the bottom of the plate at which the bottom of a plate lid rests. (Shown as *a* below.) 

Property	Description
Lid departure height	The height, in millimeters, above the bottom of the plate to which the lid is lifted. 
Lower plate at VCode	The option to lower the plate on the stage of the VCode, if the plate has a thick skirt. This allows the VCode to place the label above the thick skirt.
Can mount	The option to place the plate on top of another plate. This property is for filter plates that are placed on top of waste plates during filtration steps of a protocol.
Can be mounted	The option to place another plate on top of this plate. This property is for collection plates that collect filtrate from filter plates during the filtration steps of a protocol. Many different plates might be able to fit under any one type of filter plate. **!! IMPORTANT !! The wells of the waste plate must have a large enough diameter that the filter plate does not stick on the waste plate. The robot must be able to pick up the filter plate without the waste plate lifting up with it.**
Maximum robot handling speed	The maximum speed at which this type of plate should be moved. The general robot speed is set in VWorks. If the plate-specific robot speed (set here) is different from the general robot speed, the slower of the two speeds is used.
Length of filter tip/pin tool	The length, in millimeters of the filter tip or pin tool in filter and pin tool plates. Use a caliper to measure the length.

Related topics

For information about...	See...
Opening the labware editor	“Opening the labware editor” on page 145
Creating a labware entry (before defining it)	“Adding a labware entry” on page 147
Defining general properties	“Defining general properties” on page 152
What to do next	“Defining stacker properties” on page 158

Defining BenchCel properties

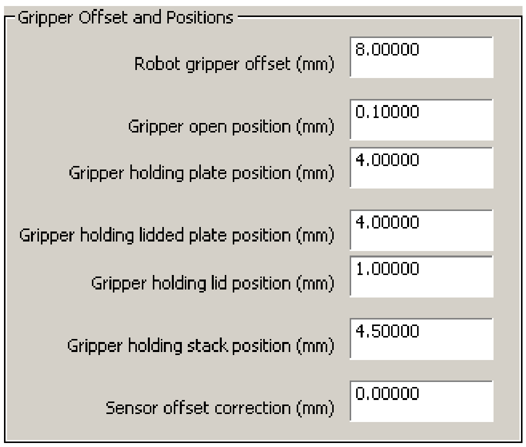
About this topic

This topic describes the properties on the BenchCel sub-tab of the Labware Editor dialog box.

Note: You must also define Stacker properties because they apply to the built-in stackers of the BenchCel.

Properties

The properties on the BenchCel sub-tab are described in the following sceenshot and table. All of the properties on this sub-tab only apply to the BenchCel.



Property	Description
Robot gripper offset	The distance, in millimeters, from the bottom of the plate to the point where the grippers grip the plate. Typically this value is 6–10 mm. *Note:* The robot gripper offset that appears on the Plate Properties page applies to Velocity11 robots used in BioCels, not BenchCel robots.
Gripper open position	The distance between the widest possible robot gripper position and the position at which the grippers are considered to be open.
Gripper holding plate position	The distance between the widest possible robot gripper position and the position at which the grippers hold a plate.
Gripper holding lidded plate position	This value is currently not used; if you enter a value, it will have no effect.

Property	Description
Gripper holding lid position	The distance between the widest possible robot gripper position and the position at which the grippers close to hold a lid.
Gripper holding stack position	The distance between the widest possible robot gripper position and the position at which the grippers close to when they are holding a stack of plates.
Sensor offset correction	The distance the plate sensor must be moved to detect a plate. Leave this value at 0 (zero).

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145
Creating a labware entry (before defining it)	"Adding a labware entry" on page 147
Defining general properties	"Defining general properties" on page 152
Defining plate properties	"Defining plate properties" on page 153
What to do next	"Defining pipette/well properties" on page 160

Defining stacker properties

About this topic

This topic describes how to define the stacker properties of the robot.

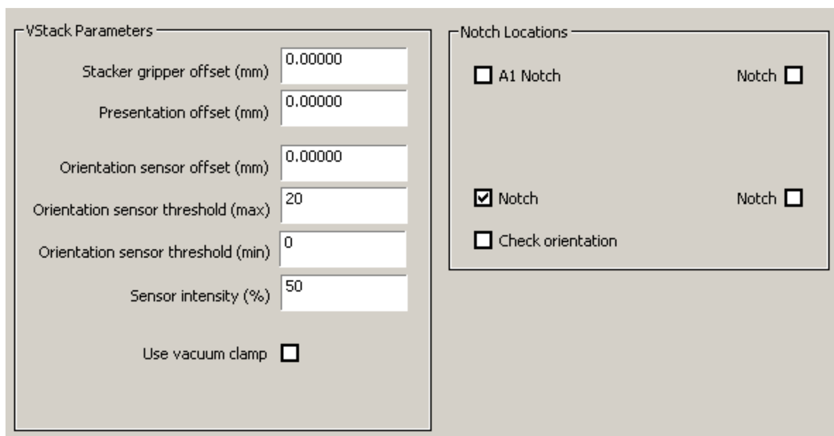
You must be logged in as an administrator or technician to perform this procedure.

Defining properties

To define stacker properties:

1. Click the **Stacker** sub-tab of the labware editor.
2. Enter the values for the available parameters according to the type of plate or labware you are defining.

   The properties on the **Stacker** sub-tab are described in the following screenshot and table.



Property	Description
Stacker gripper offset	Adjusts the height at which the plate stage stops for the grippers to grip the plate, with respect to the VStack's Grip teachpoint. Change this value only if the stacker is not gripping the plates correctly.
Presentation offset	Adjusts the height of the VStack plate stage with respect to the presentation teachpoint of a VStack.
Orientation sensor offset	Adjusts the height at which the orientation checking sensors view the plate, with respect to the VStack's sensor teachpoint. If the orientation sensor offset is 0 mm, the bottom of the plate should be in the same plane as the orientation sensors.

Property	Description
Orientation sensor threshold (max)	Determines the highest intensity of the detected light at which the stacker senses a notch when the plate stage is in the orientation sensor position. If the stacker does not sense a notch when it should, adjust the sensor threshold value. The maximum value is 255.
Orientation sensor threshold (min)	Determines the lowest intensity of the detected light at which the stacker senses a notch when the plate stage is in the orientation sensor position. If the stacker does not sense a notch when it should, adjust the sensor threshold value.
Sensor intensity	Sets the percentage of maximum sensor intensity for all sensors. If the sensor intensity is set too low, a plate will not be detected even though one is present. If it is set too high, the sensors might become saturated, causing failure to detect the orientation of a plate. This property adjusts for the fact that clear, black, and white plates reflect light differently. For example, white plates generally reflect more light so the sensor intensity should be set lower.
Use vacuum clamp	Obsolete.
Notch locations	Looks at the plate from the perspective of the robot, selects the corresponding notch or notches for your plate in the Notch Locations area.
Check orientation	Turns on plate-orientation checking.

Related topics

For information about...	See...
Opening the labware editor	“Opening the labware editor” on page 145
Creating a labware entry (before defining it)	“Adding a labware entry” on page 147
Defining general properties	“Defining general properties” on page 152
Defining plate properties	“Defining plate properties” on page 153

For information about...	See...
What to do next	"Defining pipette/well properties" on page 160

Defining pipette/well properties

About this topic

This topic describes the parameters used to define a plate or other piece of labware in the Pipette/Well Definition sub-tab of the labware editor.

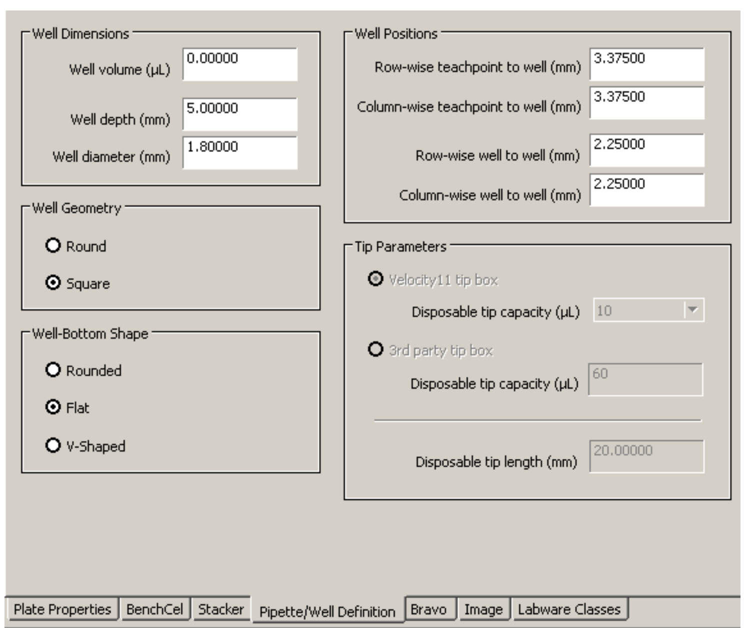
Defining properties

To define pipette/well properties:

1. Click the **Pipette/Well Definition** sub-tab of the labware editor.
2. Enter the values for the available parameters according to the type of plate or labware you are defining.

   The properties on the **Pipette/Well Definition** sub-tab are described in the following screenshot and table.

   Note: Use calipers to carefully measure the labware you are defining in the labware editor.



Property	Description
Well volume	Maximum volume of fluid for one well, in microliters.
Well depth	Distance from the top of the plate to the bottom of the well, in millimeters.
Well diameter	Diameter of the well, in millimeters.

Property	Description
Well Geometry	Shape of the wells: round or square.
Well-Bottom Shape	Shape of the well bottoms: rounded, flat, or V-shaped.
Row-wise teachpoint to well	Distance from the teachpoint to the center of the A1 well along the row (number axis), in millimeters. This setting should be 0 mm for standard 96-well plates and 2.25 mm for standard 384-well plates.
Column-wise teachpoint to well	Distance from the teachpoint to the center of the A1 well along the column (letter axis), in millimeters. This setting should be 0 mm for standard 96-well plates and 2.25 mm for standard 384-well plates.
Row-wise well to well	Distance from well-center to well-center across the row, in millimeters. This setting should be 9 mm for standard 96-well plates and 4.5 mm for standard 384-well plates.
Column-wise well to well	Distance from well-center to well-center across the column, in millimeters. This setting should be 9 mm for standard 96-well plates and 4.5 mm for standard 384-well plates.
Disposable tip capacity	Volume capacity of the disposable tips, in microliters, when labware is a tip box.
Disposable tip length	Length of the disposable tips being used, in millimeters, when labware is a tip box.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145
Creating a labware entry (before defining it)	"Adding a labware entry" on page 147
Defining general properties	"Defining general properties" on page 152
Defining plate properties	"Defining plate properties" on page 153
What to do next	"Defining Bravo properties" on page 162

Defining Bravo properties

About this topic

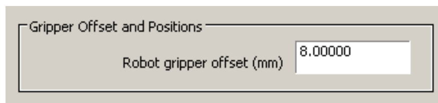
This topic describes the properties on the Bravo sub-tab of the Labware Editor dialog box.

You must be logged in as an administrator or technician to perform this procedure.

Procedure

To define Bravo properties:

1. Click the **Bravo** sub-tab of the labware editor.



2. Enter the **Robot gripper offset** (in millimeters). This is the height of the gripper above any teachpoint when the Bravo gripper is picking up or placing a plate of this type.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145
Creating a labware entry (before defining it)	"Adding a labware entry" on page 147
Defining general properties	"Defining general properties" on page 152
Defining plate properties	"Defining plate properties" on page 153
What to do next	"Inserting an image" on page 163

Inserting an image

About this topic

This topic describes how to add an image of the labware to the Image sub-tab of the labware editor.

Before you start

Make sure that the image file you want to add is in the JPG, GIF, or BMP format.

Procedure

To insert an image:

1. Open the labware editor.
2. Select the labware in the left column.
3. Click the **Image** sub-tab of the labware editor.
4. Click the ellipsis button (...), and navigate to the folder location of the image file.

5. Double-click the image file.

   The image appears below the file name.

6. Click **Save changes**.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145
Creating a labware entry (before defining it)	"Adding a labware entry" on page 147
Defining general properties	"Defining general properties" on page 152
Defining plate properties	"Defining plate properties" on page 153
Defining stacker properties	"Defining stacker properties" on page 158

For information about...	See...
Defining VPrep well properties	"Defining pipette/well properties" on page 160
What to do next	"Defining labware classes" on page 164

Defining labware classes

About this topic

This topic explains how to set up labware classes.

About labware classes

Labware classes contain labware entries. VWorks is provided with six labware classes already defined:

- ❑ Uses Filter Platepad
- ❑ Uses Standard Platepad
- ❑ Uses Taller Plate
- ❑ Uses Tip Box
- ❑ Uses Vacuum Delidder
- ❑ Uses Vacuum Platepad

Initially, you can choose to use only these supplied classes. However, if you are using an accessory, you might want to create a new class to restrict the type of labware that can be used on that accessory.

Two places to define classes

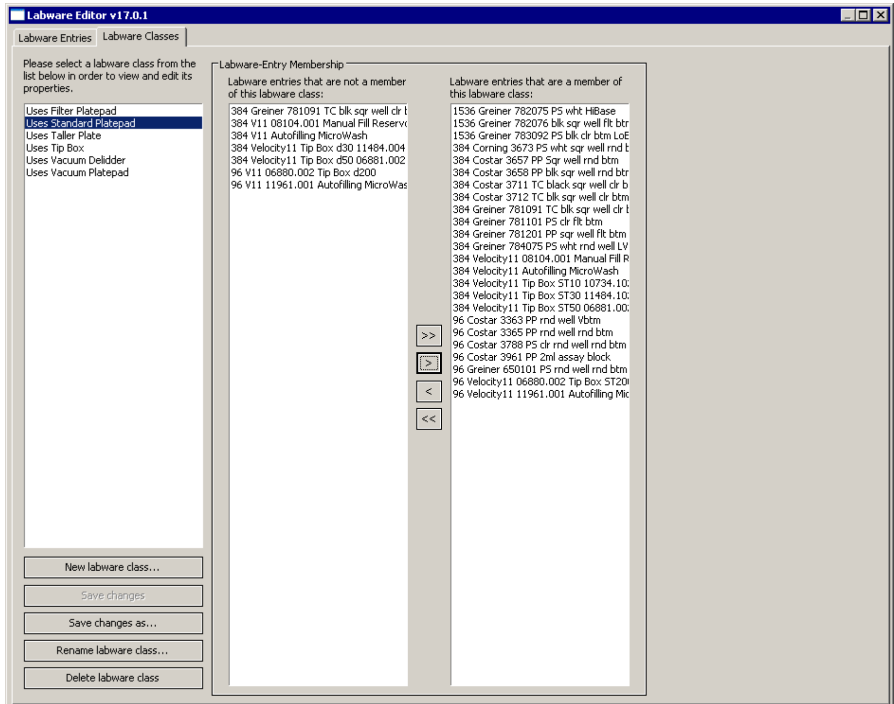
You can view and define which plate types are associated with which labware classes in:

- ❑ The Labware Classes tab
- ❑ The Labware Classes sub-tab of the Labware Entries tab

These views present the same information in different ways.

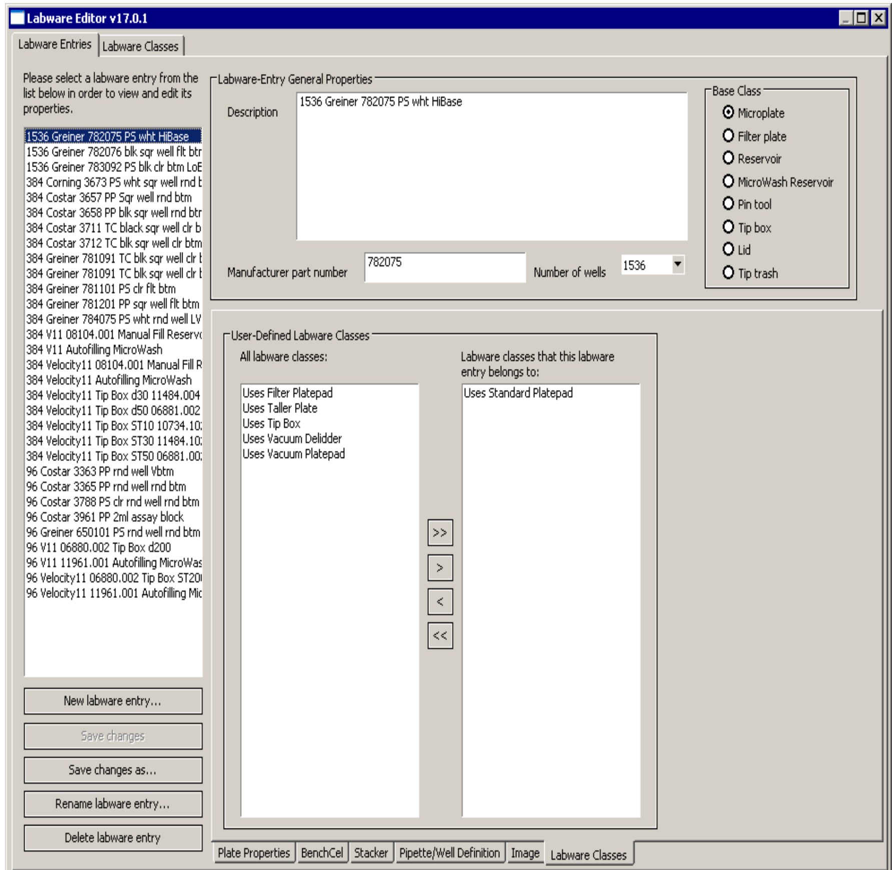
About the Labware Classes tab

In the Labware Classes tab, if you select a class in the labware selection box on the left, the plate types that are members of that class are displayed in the far right-hand column.



About the Labware Classes sub-tab

In the Labware Classes sub-page, if you select a type of plate in the labware entry box on the left, the far right-hand column displays the classes to which it is a member.



Procedure

To add a labware class:

1. Open the labware editor.
2. In the **Labware Classes** tab, click **New labware class**.
3. In the **New Labware Class** dialog box, enter a name for the labware class and click **OK**.

   The class appears in the list of labware classes.

To associate a type of plate with a labware class:

1. Open the labware editor.
2. In the **Labware Classes** tab or **Labware Classes** sub-tab, select an item from the middle column.
3. Click **>** to move the labware entries or labware classes to the right-hand column.

   To select more than one item, use SHIFT-click or CTRL + click. If you want to move all entries, click **> >**.
4. Either:
   - Click **Save Changes** to save your changes or,

- If you are in the **Labware Classes** tab, you can click **Save Changes As** to save as a new labware class.

Related topics

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 145
Creating a labware entry (before defining it)	"Adding a labware entry" on page 147
Defining general properties	"Defining general properties" on page 152
Defining plate properties	"Defining plate properties" on page 153
Defining stacker properties	"Defining stacker properties" on page 158
Defining pipette well properties	"Defining pipette/well properties" on page 160
Associating an image with a labware entry to make the entry easily recognized	"Inserting an image" on page 163

Setting liquid-handling definitions

9

The procedures in this chapter are for people with administrator or technician privileges.

This chapter contains the following topics:

- "About the liquid library editor" on page 170
- "Opening the liquid library editor" on page 172
- "Creating a liquid class" on page 173
- "Performing an accuracy calibration" on page 175

About the liquid library editor

About this topic

This topic introduces the liquid library editor, which is used to define liquid classes (pipetting properties for liquids).

Liquid library editor defined

The liquid library editor is a dialog box through which users with technician or administrator privileges can enter values for properties that affect pipetting speed, accuracy, and precision.

Default liquid library entries

When installing VWorks4, you might have elected to install the default liquid library entries. These entries are provided as examples and thus might only approximate your particular reagents. For the best performance, you should create your own liquid library definitions.

When to use the liquid library editor

You open the liquid library editor when you want to:

- ❑ View the properties that are defined for a liquid class
- ❑ Edit the properties that are defined for a liquid class
- ❑ Add new liquid classes

Liquid classes defined

The values entered into the liquid library editor can be saved as a collection, known as a liquid class. Using liquid classes saves time when writing protocols because you do not have to enter values for the liquid properties every time you create a protocol.

Types of liquid classes

You might want to create different classes for different:

- ❑ Types of liquids

  For example, water versus DMSO
- ❑ Volumes of liquids

  For example, 1 µL versus 200 µL
- ❑ Liquid operations

  For example, washing versus mixing

Liquid library database defined

The data that represents a liquid class is saved to the liquid library database, which is maintained in the Windows registry.

Using a liquid class

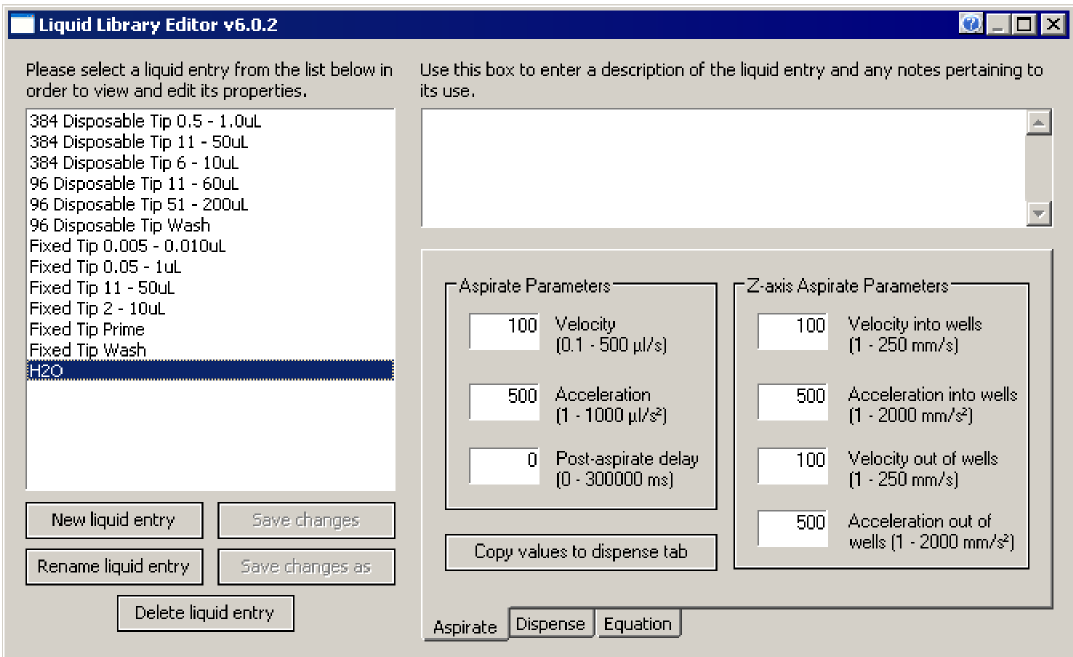
When preparing for a protocol run, you select the liquid class that you want to use. During the run, the liquid class values are referenced for pipetting operations.

Calibrating the Bravo and VPrep

The liquid library editor also has an equation editor that can be used to calibrate the Bravo and VPrep.

Liquid Library Editor dialog box

A screenshot of the liquid library editor follows.



Related topics

For information about...	See...
Opening the liquid library editor	"Opening the liquid library editor" on page 172
Creating a new liquid class entry	"Creating a liquid class" on page 173
Performing an accuracy calibration	"Performing an accuracy calibration" on page 175

Opening the liquid library editor

About this topic

This topic explains how to open the liquid library editor.

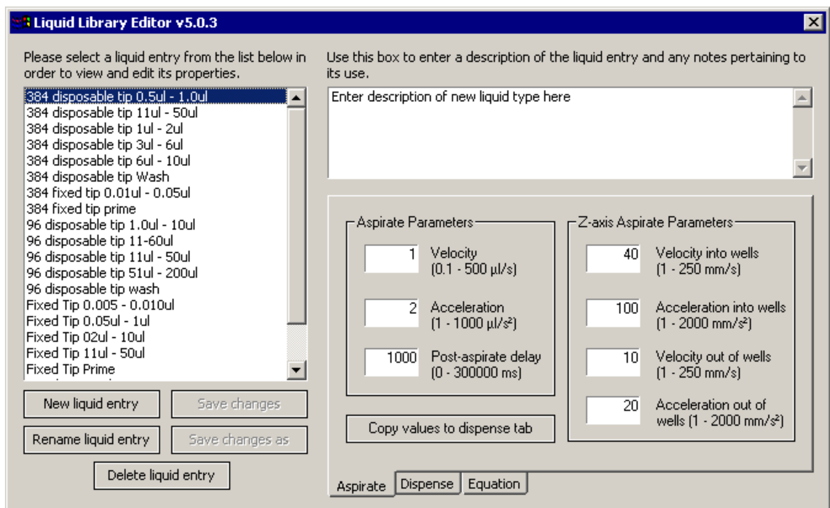
You must be logged in as an administrator or technician to open the liquid library editor.

Procedure

To open the liquid library editor from VWorks:

1. Select **Tools > Liquid Library Editor**.

   The **Liquid Library Editor** opens.



Related topics

For information about...	See...
Creating a new liquid class entry	"Creating a liquid class" on page 173
The liquid library editor	"About the liquid library editor" on page 170

Creating a liquid class

About this topic

This topic describes how to create a liquid class using the liquid library editor.

You must be logged in as an administrator or technician to perform this procedure.

Liquid compatibility

!! INJURY HAZARD !! Velocity11 products are intended to be used with non-hazardous aqueous liquids. Please contact Velocity11 before using any non-aqueous solvents or solvents generally considered to be hazardous.

Procedure

To create a liquid class:

1. Open the liquid library editor.
2. Click **New liquid entry**.
3. In the **New Liquid Entry** dialog box, enter a name for the liquid class and click **OK**.
4. Optionally, in the text box at the top right, type a note describing the liquid library entry for your records.
5. Enter values for the aspirate properties.

   The following table describes these properties.

   Note: The upper limits for some of the properties might not be achievable for the device you are using.

Aspirate property	Definition
Velocity	Specifies the speed of the aspiration stroke, in microliters per second.
Acceleration	*Bravo*. Specifies acceleration during the aspiration stroke, in microliters per second squared.
Post-aspirate delay	Specifies the time the pipettor waits after aspiration is complete before moving the tips out of the wells, in milliseconds.
Z-axis velocity into wells	Specifies how fast the pipettor moves as the tips enter the wells, in millimeters per second.
Z-axis acceleration into wells	*Bravo*. Specifies the acceleration of the pipettor as the tips move into the wells, in millimeters per second squared.
Z-axis velocity out of wells	Specifies how fast the tips leave the wells, in millimeters per second.
Z-axis acceleration out of wells	*Bravo*. Specifies the acceleration of the pipettor as the tips move out of the wells, in millimeters per second squared.

6. Click the **Dispense** tab and enter values for the dispense properties.

   The following table describes these properties.

Dispense property	Definition
Velocity	Specifies the maximum speed of the dispensing stroke, in microliters per second.
Acceleration	*Bravo*. Specifies acceleration during the dispensing stroke, in microliters per second squared.
Post-dispense delay	Specifies the time the pipettor waits after the dispense stroke before moving the tips out of the wells, in milliseconds.
Z-axis velocity into wells	Specifies how fast the pipettor moves as the tips enter the wells, in millimeters per second.
Z-axis acceleration into wells	*Bravo*. Specifies the acceleration of the pipettor as the tips enter the wells, in milliliters per second squared.
Z-axis velocity out of wells	Specifies how fast the pipettor moves as the tips leave the wells, in millimeters per second.
Z-axis acceleration out of wells	*Bravo*. Specifies the acceleration of the pipettor as the tips leave the wells, in millimeters per second squared.

7. Click **Save changes**.

   The changes are now stored in the liquid library database.

Related topics

For information about...	See...
Opening the liquid library editor	"Opening the liquid library editor" on page 172
Calibrating a pipettor	"Performing an accuracy calibration" on page 175
The liquid library editor	"About the liquid library editor" on page 170

Performing an accuracy calibration

About this topic

This topic describes how to improve the accuracy of pipetted volumes by:

- ❑ Calibrating the pipettor
- ❑ Plotting the actual volume dispensed as a function of the set dispense volume
- ❑ Calculating the polynomial coefficients of the plot
- ❑ Entering the coefficients into the liquid library equation editor

Do you need to calibrate your pipettor?

Pipetting accuracy is the ability to dispense an absolute volume of liquid. In practice, the volume that is actually dispensed by a pipettor may be different from the dispense volume that you select. This difference is the absolute error.

In some protocols, as long as you dispense an excess of liquid, the actual volume pipetted is not important. In other protocols, pipetting accuracy can be a critical factor. You must remember, though, that every step of an experiment has error and there is no point taking the time to improve the accuracy of pipetting to four significant digits if another step in your protocol has error at the third significant digit.

If you are sure that the overall error of the experiment is limited by pipetting accuracy, and error at this number of significant figures makes a practical difference to your interpretation of the data, consider performing an accuracy calibration.

Method overview

This section gives an overview of the method you can use to measure pipetting accuracy. It does not give a detailed procedure because that depends on exactly how you choose to conduct the experiment.

To calibrate a pipettor, an independent method of measuring dispensed volume is required. One method is to dispense a solution of fluorescein dye and measure the fluorescence emitted from each plate well.

!! IMPORTANT !! Whichever method you use, verify that the error of the detection method is significantly smaller than the pipetting error. Otherwise, the error you detect might be from the detection method and not the pipetting error.

The overall method is:

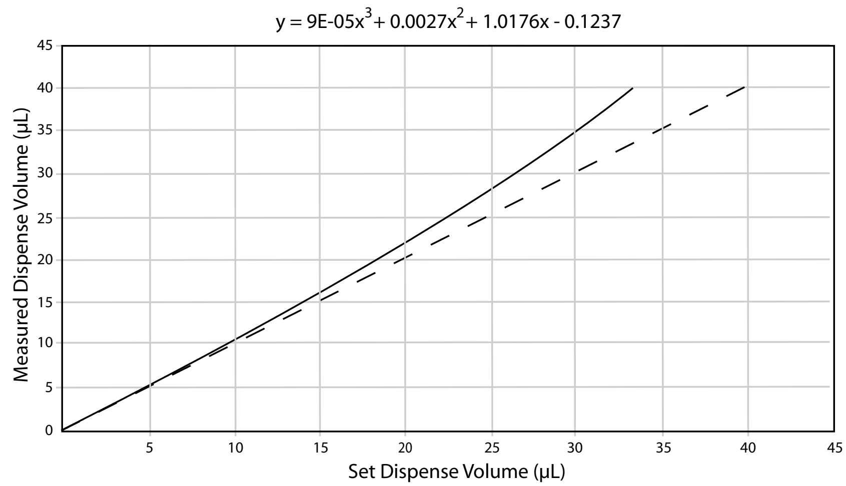
1. Perform a series of pipetting operations in which different volumes are pipetted.
2. Measure the volumes of dispensed liquid using the independent measuring method.
3. In a spreadsheet program, tabulate the dispense volumes that you set in the software against the measured volumes.

4. Plot a graph, with the set dispense volume on the *y*-axis and measured dispense volume on the *x*-axis.

   The plot will be a curve, reflecting the fact that absolute error is a function of the magnitude of the measurement.

5. Use the statistical functions of the spreadsheet program to fit a curve to the data.

   Your result might look like this:



   The dashed line is a reference line, where the set dispense volume equals the measured dispense volume. The equation is the polynomial for the line, calculated by the spreadsheet program.

6. Enter the curve information into the equation editor of the liquid library editor.

If you repeat the experiment, you will find that the curve is much closer to a straight line. This is because the equation you entered adjusts the action of the servo motor that determines aspirate and dispense volumes, thereby calibrating the dispense.

Using the equation editor

The equation editor in the liquid library editor is where you enter the calibration curve data to correct for pipetting inaccuracy.

To enter a polynomial into the equation editor:

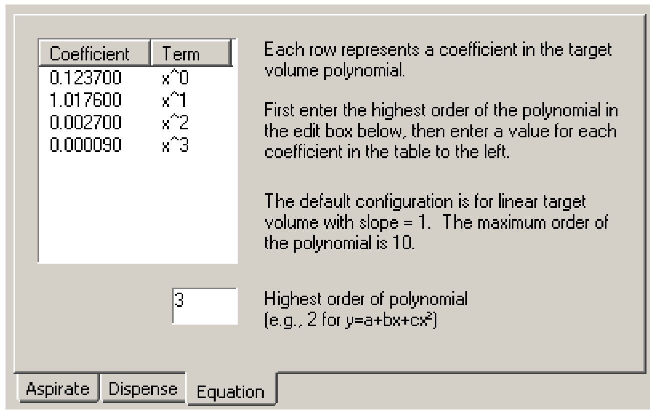
1. Open the liquid library editor.
2. Click the **Equation** tab to display the equation editor.
3. In the **Highest order of polynomial** text box, enter the value for the highest order of the polynomial.

   This is the largest exponent in the equation and tells you how many terms are in the equation. For example, if the highest order of the polynomial is 3, the equation will have the general form: $y = a + bx + cx^2 + dx^3$, where ‘x’ is the volume specified by any pipettor task that uses this liquid class. With an exponent of three, four rows are added to the equation editor table.

4. In the Coefficient/Term table, in turn, enter the coefficient and exponent for each of the terms in the equation, starting with the zero order term.

   To enter a value, single-click the **Coefficient** table row twice. Note that the exponents are already entered for you and cannot be edited.

   The following example is for the curve displayed in the previous graph.



5. Click **Save changes**.

Administrator procedures

10

The procedures in this chapter are for people with administrator or technician privileges.

This chapter contains the following topics:

- ❑ “About user accounts and privileges” on page 180
- ❑ “Adding and deleting a user account” on page 181
- ❑ “Moving or sending a registry file” on page 183

About user accounts and privileges

About this topic

You must have a user account to log in to VWorks. Your user account is associated with a user role that determines the privileges you have to perform particular functions.

This topic describes the privileges associated with different user roles.

The effect of privileges

Privileges have the following effects:

- If you do not have the privilege to perform a function associated with a particular menu command, the text of the command is gray.
- If you do not have the privilege to perform the functions accessed from a particular tabbed page, the tab is not visible to you.
- In some cases, if you do not have the privilege to perform an operation, when you attempt the operation you get an error message telling you that your privileges are insufficient.

User roles and privileges

User roles enforce the following privileges:

User role	Has privileges to...
Guest	Run existing protocols.
Operator	❑ Perform guest functions (see above). ❑ Operate devices in real-time using diagnostics software.
Technician	❑ Perform operator functions (see above). ❑ Create and save protocols. ❑ Edit the labware database and liquid library database.
Administrator	❑ Perform technician functions (see above). ❑ Manage devices through the device manager. ❑ Create and delete user accounts. ❑ Run a protocol that contains compiler errors.

Related topics

For more information about...	See...
Moving or sending a registry file	"Moving or sending a registry file" on page 183
Adding and deleting a user account	"Adding and deleting a user account" on page 181
Setting up email	"Notification of errors by email" on page 32

Adding and deleting a user account

About this topic

We recommend that VWorks administrators create an account for every user. The privileges set for the account should be appropriate for the user's job role.

This topic explains how to add and delete user accounts.

About user accounts and passwords

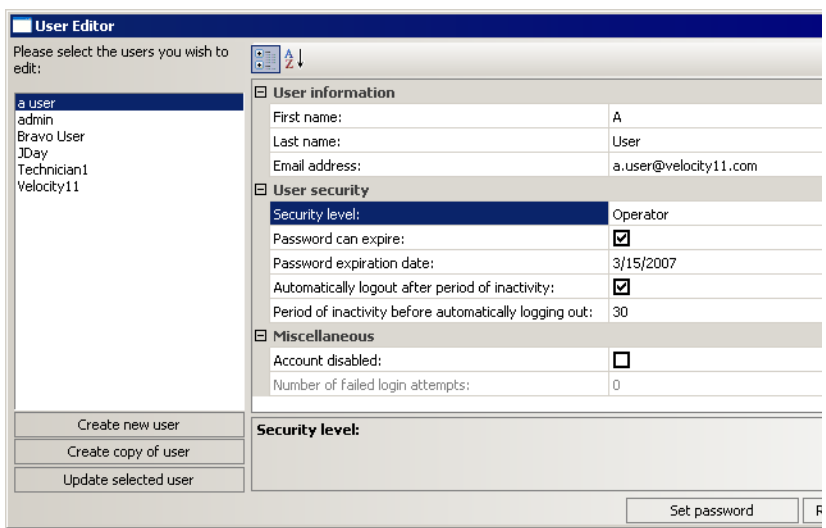
User accounts and passwords use the following conventions:

- User accounts can be disabled but not deleted.
- User names and passwords are case-sensitive.
- Passwords must contain six or more characters.
- If a user enters an incorrect password three times consecutively, the user is locked out until an administrator resets the account.

Adding a user account

To add a user account:

1. Select **Tools** > **User Management**.
2. In the **User Editor** dialog box, click **Create new user** and enter a name for the user.
3. Enter values in the **User Editor** dialog box.



4. Click **Update selected user** to save changes, and then close the dialog box.

Disabling a user account

You cannot delete a user account, but you can disable it.

To disable an account so that the user cannot log in:

1. Select **Tools** > **User Management**.
2. Select the account.

3. Select the **Account disabled** check box.
4. Click **Update selected user** to save changes, and then close the dialog box.

Resetting a user account

If a user mistypes the password three consecutive times, the user is locked out of the account until an administrator resets the account.

To reset an account:

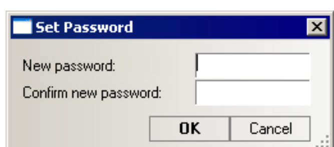
1. Select **Tools** > **User Management**.
2. Select the account.
3. Click **Reset failed login count**.
4. Click **Update selected user** to save changes, and then close the dialog box.

Changing a password

An administrator can reset the password of any account. Operators and technicians can change their own passwords.

To reset a password:

1. Select **Tools** > **User Management**.
2. Select the account.
3. Click **Set password**, enter the new password twice and click **OK**.



4. Close the dialog box.

Related topics

For more information about...	See...
Moving or sending a registry file	“Moving or sending a registry file” on page 183
Setting up email	“Notification of errors by email” on page 32

Moving or sending a registry file

About this topic

This topic provides instructions on how to export a Windows registry file for import to another computer or for emailing to Velocity11.

When to do this

You might need to copy or send a registry file in the following situations:

- ❑ To move a labware or liquid library database to other devices using a different controlling computer
- ❑ To make a backup of a VWorks profile
- ❑ To transfer a VWorks profile from one computer to another
- ❑ To email a labware or liquid library database or VWorks profile when requested by personnel at Velocity11

About moving data

The labware and liquid libraries and profiles are maintained in the Windows registry of the controlling computer.

When you use VWorks to make a change to any of these, the change is seen when accessing that information from the networked device's software. This is because VWorks and device software run on the same controlling computer and access the same databases.

If you make a change to the labware or liquids database or a profile, you can use a two-step process to propagate the change to another computer.

1. Export the Windows registry key containing the data to a file.
2. Import the file to the other computer's registry.

Damage hazard

!! DAMAGE HAZARD !! Making a mistake when editing the registry might cause critical failures with your operating system.

Exporting a registry key

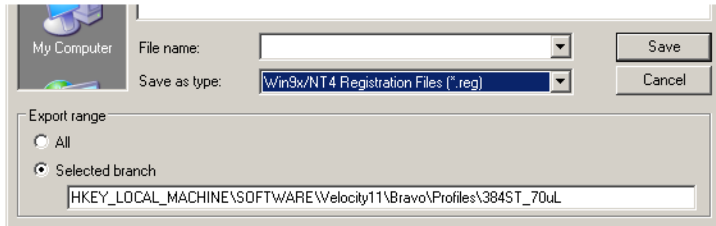
To export a registry key:

1. From the Windows **Start** menu, select **Run**.
2. In the **Open** text box, type `regedit`.
3. Click **OK**.

   The Windows registry editor opens.
4. Expand folders to display and select one of the following folders:
   - ◆ HKEY_LOCAL_MACHINE\SOFTWARE\Velocity11\Shared\Labware
   - ◆ HKEY_LOCAL_MACHINE\SOFTWARE\Velocity11\Shared\Liquid Library
   - ◆ HKEY_LOCAL_MACHINE\SOFTWARE\Velocity11\VWorks\Profiles
5. From the **Registry** (or **File**) menu, select **Export**.

   The **Export Registry File** browser box opens.

6. Before saving the file, make sure you:
   - Select **Selected branch**.
   - If you are moving the file to a computer with a different Windows operating system, set **Save as type** appropriately.



7. Save the file.
8. Select **Registry** > **Exit** (or **File** > **Exit)** to close the registry editor.

Importing a registry key

If this is the first time you are importing a registry file to the computer, you need to use the Open With command.

Before you start

You must have Windows Administrator permissions to perform this task.

To import a registry key:

1. Copy the registry file to any location on the recipient computer.
2. On the recipient computer, double-click the registry file.

The information in the file is written automatically to the registry.

Emailing a registry file

Occasionally, you might be asked to send a registry file to Velocity11.

To email a registry file:

1. Export the Windows registry key containing the data to a file.
2. Change the file's extension to .re_ (This is necessary because many email servers do not allow *.reg files to be emailed.)
3. Email the file.

Related topics

For more information about...	See...
Adding and deleting a user account	"Adding and deleting a user account" on page 181
Setting up email	"Notification of errors by email" on page 32

Maintenance and troubleshooting

11

This chapter contains the following topics:

- ❑ "About error handling" on page 186
- ❑ "Compilation warnings and errors" on page 187
- ❑ "Resolving device initialization errors" on page 188
- ❑ "Importing a log file to Excel" on page 190

About error handling

About this topic

VWorks is complex software that requires the interaction of hundreds of parameters, configurations, and operator setup tasks for a protocol to run successfully.

Causes of errors

Errors have many causes, including poor protocol writing, incorrect operator setup, variation in plates, hardware failure, and software failure.

Handling errors during normal operation

Error handling is a normal part of operating a lab automation system and that when errors occur it does not necessarily mean that your lab automation system has malfunctioned.

Related topics

For information about...	See...
Sending a bug report to Velocity11	"Sending a bug report" on page 8
Resolving device initialization errors	"Resolving device initialization errors" on page 188
Protocol compiling errors	"Compilation warnings and errors" on page 187

Compilation warnings and errors

About this topic

This topic briefly describes compilation warnings and errors.

Compilation warnings

Warnings alert you to situations that seem to have incorrect intent but that will not cause a collision or a device to fail.

Compilation warnings are displayed in the VWorks log toolbar, listing the task that caused the warning.

Warnings are generated, for example, when:

- Pipette tips are on the VPrep head at the start of the protocol so a tips-off task is run before a tips-on task
- No label is printed during a VCode task

Compilation errors

Errors alert you to situations where a protocol or device will fail.

Compilation errors are listed in the Log toolbar, and a dialog box opens, telling you how many errors were found. If you are logged in with technician, operator, or guest privileges, you are unable to continue with the protocol. If you are logged in with administrator privileges, the dialog box allows you to run the protocol despite the errors.

Errors are generated when:

- Operating parameters are out of range, denoted by red text in fields of the Task Parameters toolbar
- A task asks a VPrep to pipette from a plate that does not exist in the system
- Volumes in pipette steps do not match, such as when a dispense volume is greater than a previous aspirate volume
- A specific stacker has not been assigned to a stacker task
- A Signal task has no associated Waitfor task

Related topics

For information about...	See...
Using the Log toolbar	"Working with the Log toolbars" on page 44
Using simulation mode	"Simulating a run" on page 84
Resolving deadlocks	"About setting the number of simultaneous plates" on page 79

Resolving device initialization errors

About this topic

This topic describes the device initialization process and suggests some things you can try if you encounter problems with device initialization.

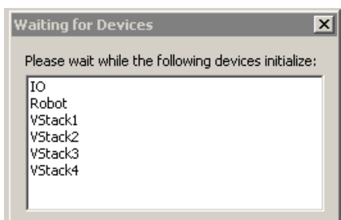
Device initialization process

When you start VWorks, the software loads the driver files for all the devices on your system. A record of this process is displayed in the log toolbar.

When you open a protocol file, the device file associated with that protocol opens. The device file tells the software which devices are connected to the system. Then, for some kinds of devices, an initialization step tests the communication between VWorks and the device.

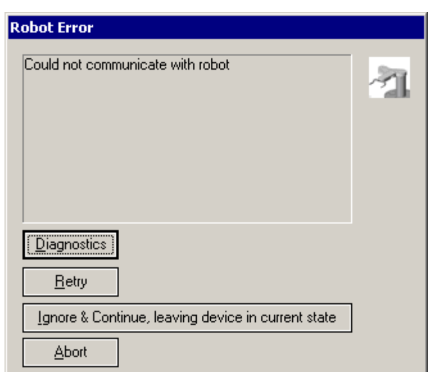
Resolving initialization errors

During this process a message window opens, displaying the name of the devices that the software expects to find. Devices are removed from the list as the system computer determines that the devices are ready. In the following example, there are six devices that the software is trying to initialize:



If there is a problem with initializing a device, you receive an error message dialog box for that device. The problem encountered is stated in the text field of the dialog box.

In this example, VWorks could not initialize the Robot device.



Make sure the device is turned on and that the communications cable is connected properly.

Click **Diagnostics** and try to resolve the problem in the device profile. If that does not solve the problem, contact the Velocity11 Service Center.

Related topics

For information about...	See...
Using diagnostics for a device	*Device Driver User Guide*
Using Bravo Diagnostics	*Bravo User Guide*

Importing a log file to Excel

About this topic

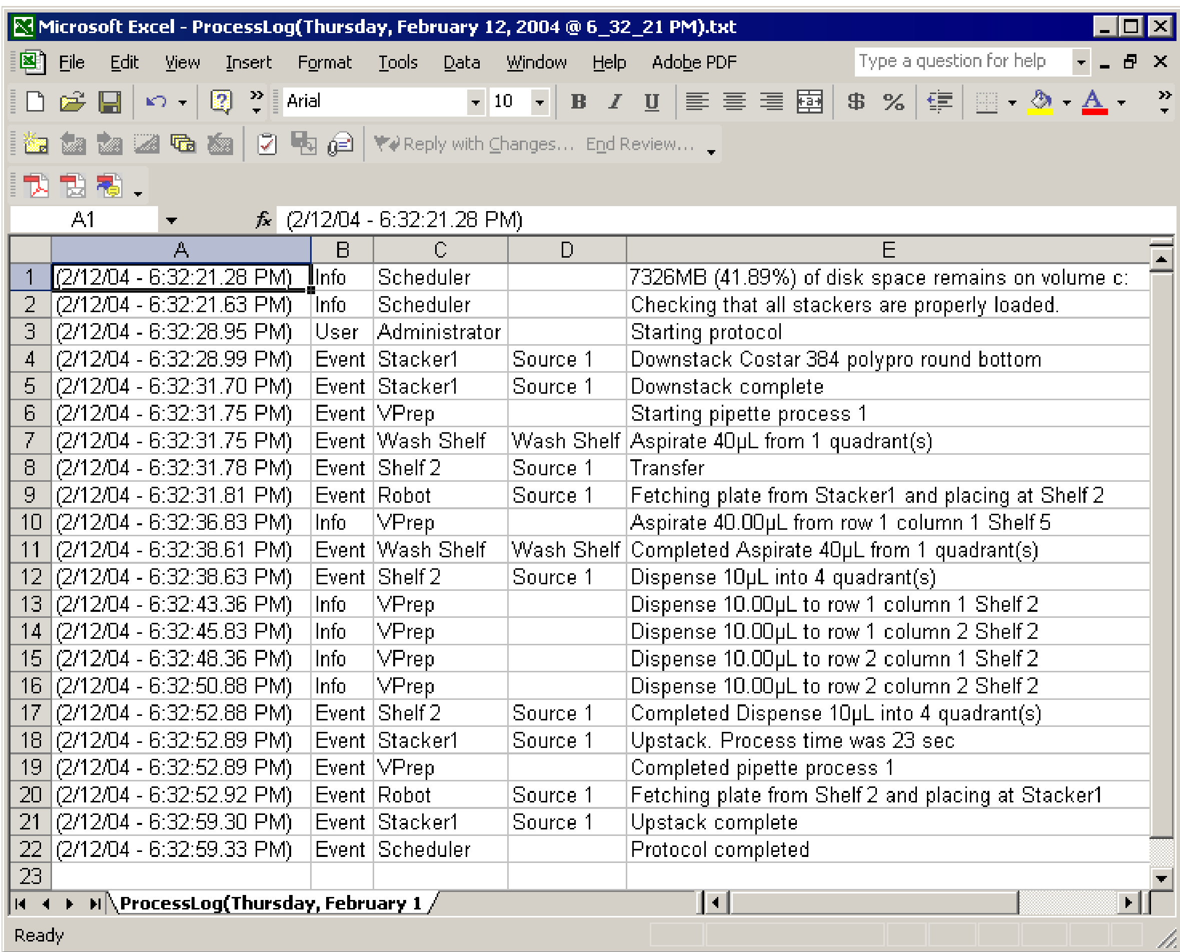
This topic describes how to import a log file into Microsoft Excel. Log file data can then be reviewed and analyzed in Excel.

Procedure

To import a log file to Excel:

1. Open Microsoft Excel.
2. Drag the file onto the Excel window.

   The data is imported.



	A	B	C	D	E
1	(2/12/04 - 6:32:21.28 PM)	Info	Scheduler		7326MB (41.89%) of disk space remains on volume c:
2	(2/12/04 - 6:32:21.63 PM)	Info	Scheduler		Checking that all stackers are properly loaded.
3	(2/12/04 - 6:32:28.95 PM)	User	Administrator		Starting protocol
4	(2/12/04 - 6:32:28.99 PM)	Event	Stacker1	Source 1	Downstack Costar 384 polypro round bottom
5	(2/12/04 - 6:32:31.70 PM)	Event	Stacker1	Source 1	Downstack complete
6	(2/12/04 - 6:32:31.75 PM)	Event	VPrep		Starting pipette process 1
7	(2/12/04 - 6:32:31.75 PM)	Event	Wash Shelf	Wash Shelf	Aspirate 40µL from 1 quadrant(s)
8	(2/12/04 - 6:32:31.78 PM)	Event	Shelf 2	Source 1	Transfer
9	(2/12/04 - 6:32:31.81 PM)	Event	Robot	Source 1	Fetching plate from Stacker1 and placing at Shelf 2
10	(2/12/04 - 6:32:36.83 PM)	Info	VPrep		Aspirate 40.00µL from row 1 column 1 Shelf 5
11	(2/12/04 - 6:32:38.61 PM)	Event	Wash Shelf	Wash Shelf	Completed Aspirate 40µL from 1 quadrant(s)
12	(2/12/04 - 6:32:38.63 PM)	Event	Shelf 2	Source 1	Dispense 10µL into 4 quadrant(s)
13	(2/12/04 - 6:32:43.36 PM)	Info	VPrep		Dispense 10.00µL to row 1 column 1 Shelf 2
14	(2/12/04 - 6:32:45.83 PM)	Info	VPrep		Dispense 10.00µL to row 1 column 2 Shelf 2
15	(2/12/04 - 6:32:48.36 PM)	Info	VPrep		Dispense 10.00µL to row 2 column 1 Shelf 2
16	(2/12/04 - 6:32:50.88 PM)	Info	VPrep		Dispense 10.00µL to row 2 column 2 Shelf 2
17	(2/12/04 - 6:32:52.88 PM)	Event	Shelf 2	Source 1	Completed Dispense 10µL into 4 quadrant(s)
18	(2/12/04 - 6:32:52.89 PM)	Event	Stacker1	Source 1	Upstack. Process time was 23 sec
19	(2/12/04 - 6:32:52.89 PM)	Event	VPrep		Completed pipette process 1
20	(2/12/04 - 6:32:52.92 PM)	Event	Robot	Source 1	Fetching plate from Shelf 2 and placing at Stacker1
21	(2/12/04 - 6:32:59.30 PM)	Event	Stacker1	Source 1	Upstack complete
22	(2/12/04 - 6:32:59.33 PM)	Event	Scheduler		Protocol completed
23					

Protocol log file information

The information given in a protocol log file is explained in the following table. Refer to the screenshot of the Excel file for the column letters.

Spreadsheet column	Information
A	Date and time that the entry was added

Spreadsheet column	Information
B	Type of information: ❑ Error ❑ Event ❑ Info ❑ Script ❑ User
C	Origin of the information
D	Either the: ❑ Name of the plate ❑ Name of the device, if it refers to a reagent
E	Description of the log entry

Related topics

For information about...	See...
Setting log file directories	“Setting log file directories” on page 34
Reporting a problem to Velocity11	“Reporting VWorks problems” on page 7
Working with the Log toolbar	“Working with the Log toolbars” on page 44

Index

Note: You can search our technical documentation on our website at www.velocity11.com.

Symbols
.dev file format, *25*
.pro file format, *25*

Numerics
21 CFR Part 11 compliance, *12*

A
aborting a run, *41*
acceleration property, *173*, *174*
accuracy calibration, *175*
Acrobat Reader, *5*
Acrobat viewer, *5*
administrator privilege, *180*
Advanced Settings
    JavaScript, *135*
    Task Parameters toolbar, *21*
    task time estimate, *132*
agitate task. see Shake (Bravo) task, *118*
Aspirate task
    defined, *96*
    setting parameters, *96*
Available Devices toolbar, *18*
Available Tasks toolbar, *18*

B
bar code file, tracking plates, *86*
Bar codes Directory, *75*
BenchCel
    defining labware properties, *156*
    device, *13*
Bravo
    defining labware properties, *162*
    device, *13*
    setting protocol options, *75*
    stand-alone, *53*
    task parameters, *90*

C
calibration accuracy, *175*
can be mounted property, *155*
can be sealed? property, *154*
can have lid? property, *154*
can mount property, *155*
check orientation property, *159*
class, labware, *140*
cleanup protocol, *17*, *80*
cleanup protocol process
    defined, *69*
    setting up, *80*
column-wise teachpoint to well property, *161*
column-wise well to well property, *161*
compiling
    a protocol, *83*
    errors, *187*
    warnings, *187*
computer connection options, *51*
Configure Labware task
    defined, *92*
    setting parameters, *92*
Control toolbar, *22*
copying a labware entry, *151*
crossover cable, *59*

D
database
    copying, *183*
    labware, *183*
    liquid library, *183*
deadlock, *79*, *84*
default user account, password for, *30*
deleting labware entries, *149*
device files, *24*
    choosing for a protocol, *75*
    workspace, *16*
Device Selection, *20*, *131*
devices
    for select task, *20*
    initialization errors, *188*
    list of, *13*
DHCP server
    defined, *61*
    initialization file, *53*
    installing, *61*
    kit, *54*
    stopping and starting, *62*
DHCP Server Kit, *61*
diagnostics version number, *6*
DiscoveredBioNet Devices dialog box, *63*
Dispense task
    defined, *99*
    setting parameters, *99*
disposable tip capacity property, *161*
disposable tip length property, *161*

E

email
- notification of errors by, *32*
- setting up, *32*

equation editor, using, *176*
errors
- about, *186*
- causes, about, *186*
- notification of, *32*
- reporting, *7*
- troubleshooting initialization, *188*

Ethernet
- choosing as connection, *51*
- connection workflow for, *55*
- firewall, *63*
- installing a direct connection, *59*
- networking requirements, *52*
- switch, *59*

F

file format for protocol, *71*
Find button, *5*
finding the file, *4*
firewall
- defined, *63*
- passing data, *64*
- program, *63*
- Windows XP, *63*

first login, *30*

G

general properties for labware, defining, *152*
gripper holding lid position property, *157*
gripper holding lidded plate position property, *156*
gripper holding plate position property, *156*
gripper holding stack position property, *157*
gripper offset Bravo property, *162*
gripper open position property, *156*
guest privilege, *180*

I

image, inserting labware, *163*
installing VWorks, *48*, *50*
instruments, list of, *13*
IP address
- about, *52*
- assigning dynamically, *61*
- checking, *60*
- configuring computer's, *56*
- dynamic, *52*, *63*
- static, *52*, *53*, *63*

J

JavaScript
- about, *134*
- example, *137*
- properties, *136*
- resources, *134*
- task object, *136*
- where to write, *135*

job roles for readers of this guide, *2*

L

labware
- adding, *147*
- associating plate with a class, *166*
- classes page, *164*
- classes, about, *140*
- copying, *151*
- defining BenchCel properties, *156*
- defining Bravo properties, *162*
- defining classes, *164*
- defining general properties, *152*
- defining pipette/well properties, *160*
- defining plate properties, *153*
- defining stacker properties, *158*
- deleting, *149*
- editor, about, *140*
- inserting an image, *163*
- renaming, *150*

labware classes sub-page, *165*
labware database, *24*, *183*
labware editor
- classes page, *143*
- entries page, *142*
- opening, *145*

length of filter tip/pin tool property, *155*
license, *3*, *12*
- Benchtop, *12*
- Instrument, *12*

lid departure height property, *155*
lid gripper offset property, *154*
lid resting height property, *154*
lidded stacking thickness property, *154*
lidded thickness property, *154*
liquid library, *24*
- adding an entry, *173*
- calibrating VPrep, *170*
- changing database, *183*
- classes defined, *170*
- database defined, *170*
- editor defined, *170*
- opening the editor, *172*

liquid library editor
- about, *171*

opening, *172*
location properties
defined, *94*
set for Benchtop license, *94*
set for Instrument license, *94*
log file
changing location, *34*
importing into Excel, *190*
protocol, *190*
log toolbars, *40*, *44*
logging in to VWorks, *30*
lower plate at VCode property, *155*

M

Main Log toolbar, *19*, *44*
maximum robot handling speed property, *155*
menus, overview, *22*
Mix task
defined, *102*
setting parameters, *102*
mobile phone, notification of errors, *32*
Move To Location task
defined, *105*
setting parameters, *105*
moving toolbars, *15*

N

network
integrating into existing, *53*
setting up new, *53*
network card, *52*
checking IP address, *60*
configuring, *56*
networking
connecting to a LAN, *59*
multiple VCodes, *60*
safety, *53*
see also Ethernet
networking kit. *see* DHCP server kit
notch locations property, *159*

O

og, *44*
olbars, *44*
online help, *4*
opening
labware editor, *145*
liquid library editor, *172*
protocols in VWorks, *31*
operator privilege, *180*
options for log files, *34*
orientation sensor offset property, *158*
orientation sensor threshold (max) property, *159*
orientation sensor threshold (min) property, *159*

P

pager, notification of errors by, *32*
passwords
about, *181*
changing, *30*, *181*
first login, *30*
pausing a run, *41*
PDF guide, *4*, *5*
Pipette Log toolbar, *19*, *44*
pipette/well, defining labware properties, *160*
pipetting accuracy, *175*
plate icon, setting parameters, *77*
plate properties, defining, *153*
PlateLoc device, *13*
PlatePad device, *13*
plates, tracking with bar codes, *86*
post-aspirate delay property, *173*
post-dispense delay property, *174*
preparing runs, workflow for, *28*
presentation offset property, *158*
privileges
defined, *180*
effect of, *180*
types, *180*
profile, defined, *24*
Progress toolbar, *20*
property
acceleration, *173*, *174*
can be mounted, *155*
can be sealed?, *154*
can have lid?, *154*
can mount, *155*
check orientation, *159*
column-wise teachpoint to well, *161*
column-wise well to well, *161*
disposable tip capacity, *161*
disposable tip length, *161*
gripper holding lid position, *157*
gripper holding lidded plate position, *156*
gripper holding plate position, *156*
gripper holding stack position, *157*
gripper offset, *162*
gripper open position, *156*
length of filter tip/pin tool, *155*
lid departure height, *155*
lid gripper offset, *154*
lid resting height, *154*
lidded stacking thickness, *154*
lidded thickness, *154*
lower plate at VCode, *155*
maximum robot handling speed, *155*

notch locations, *159*
orientation sensor offset, *158*
orientation sensor threshold, *159*
post-aspirate delay, *173*
post-dispense delay, *174*
presentation offset, *158*
robot gripper offset, *156*
row-wise teachpoint to well, *161*
row-wise well to well, *161*
sealed stacking thickness, *154*
sealed thickness, *154*
sensor intensity, *159*
sensor offset correction, *157*
stacker gripper offset, *158*
stacking thickness, *154*
thickness, *153*
use vacuum clamps, *159*
velocity, *173*, *174*
well depth, *160*
well diameter, *160*
well geometry, *161*
well volume, *160*
well-bottom shape, *161*
z-axis acceleration into wells, *173*, *174*
z-axis acceleration out of wells, *173*, *174*
z-axis speed into wells, *173*
z-axis velocity into wells, *174*
z-axis velocity out of wells, *173*, *174*
protocol
adding notes about, *75*
bar code file, *86*
cleanup, *17*, *80*
compiling, *83*
creating, *74*
creating, workflow for, *73*
defined, *69*
example of, *72*
file format, *71*
filename, *25*
files, overview, *17*
log file example, *190*
opening in VWorks, *31*
options, *17*
saving, *84*
setting options, *75*
simulating, *84*
startup, *17*, *80*
what you should know, *35*
protocol alias, *75*
Pump Reagent task
defined, *106*
setting parameters, *106*

R

registry key
emailing, *184*
exporting, *183*
importing, *184*
renaming a labware entry, *150*
reporting errors, *7*
Reserve Location task
defined, *108*
setting parameters, *108*
Robot gripper offset property, *156*
robots, supported types, *13*
row-wise teachpoint to well property, *161*
row-wise well to well property, *161*
run
aborting, *41*
cleaning up after, *46*
defined, *69*
monitoring, *43*
overview, *38*
starting, *39*
what happens after, *40*

S

saving a protocol, *84*
screen message options, *44*
sealed stacking thickness property, *154*
sealed thickness property, *154*
sensor intensity property, *159*
sensor offset correction property, *157*
serial connection, choosing, *51*
Serial Dilution task
compatible head and plate, *110*
configuring labware, *110*
defined, *110*
out-of-bounds locations, *109*
pipette head requirements, *109*
setting parameters, *111*
setting the head mode, *111*
using the wizard, *112*
Set Head Mode task
defined, *117*
setting parameters, *117*
Shake (Bravo) task
defined, *118*
setting parameters, *118*
simulating a protocol, *84*
simultaneous plates, *78*, *79*
stacker gripper
offset property, *158*
stacker, defining properties for labware, *158*
stacking thickness property, *154*
stand-alone connection, *53*

Standard toolbar, *21*
starting
    runs, *39*
    VWorks, *29*
startup protocol, *17*, *80*
startup protocol process
    setting up, *80*
status bar, *23*
stopping a run, *41*
Sub Process (Bravo) task
    defined, *119*
    setting parameters, *119*
sub-process, defined, *68*

T

Task Parameters toolbar, *20*
task parameters, for Bravo, *90*
tasks
    adding, *81*
    defined, *68*
    deleting, *82*
    moving, *82*
    time estimate, *132*
    types, *90*
teachpoint file, *24*, *25*
technician privilege, *180*
text message, notification of errors, *32*
thickness property, *153*
time estimate for tasks, *21*, *132*
Tips Off (Bravo) task
    defined, *121*
    setting parameters, *121*
Tips On (Bravo) task
    defined, *120*
    setting parameters, *120*
toolbars, *14*
    Available Devices, *18*
    Available Tasks, *18*
    Control, *22*
    Main Log, *19*, *44*
    moving, *15*
    Pipette Log, *19*, *44*
    Progress, *20*
    Standard, *21*
    Task Parameters, *20*
    Workspace, *17*
tracking plates with bar codes, *86*
troubleshooting initialization errors, *188*

U

uninstalling VWorks, *65*
use vacuum clamp property, *159*
user accounts
    adding, *181*
    changing password, *182*
    disabling, *181*
    editing, *181*
    passwords
        resetting, *181*
    privileges, *180*
    resetting lock out, *182*
user database, *24*
user interface
    controlling appearance of, *14*
    overview, *14*
User Message task, *41*
    defined, *123*
    setting parameters, *123*

V

Vacuum filter (Bravo) task
    defined, *124*
    setting parameters, *124*
velocity property, *173*, *174*
VWorks
    finding the version number, *6*
    installing, *48*, *50*
    introduction to, *12*
    license types, *12*
    logging in, *30*
    starting, *29*
    uninstalling, *65*
    user interface, overview, *14*
    version number, *6*

W

Wait For User (Bravo) task
    defined, *126*
    example, *126*
Wash Tips (Bravo) task
    defined, *127*
    setting parameters, *127*
well depth property, *160*
well diameter property, *160*
well geometry property, *161*
well volume property, *160*
well-bottom shape property, *161*
Windows XP, *63*
Windows, registry key, copying, *183*
Workspace toolbar, *17*

Z

z-axis acceleration into wells property, *173*, *174*
z-axis acceleration out of wells property, *173*, *174*
z-axis speed into wells property, *173*
z-axis velocity into wells property, *174*

z-axis velocity out of wells property, *173*, *174*

Agilent Technologies

User Guide

G5415-90000