



PolySpace™
Installation Guide R2007a+





How to Contact The MathWorks

www.mathworks.com
comp.soft-sys.matlab
www.mathworks.com/contact_TS.html

Web
Newsgroup
Technical Support

suggest@mathworks.com
bugs@mathworks.com
doc@mathworks.com
service@mathworks.com
info@mathworks.com

Product enhancement suggestions
Bug reports
Documentation error reports
Order status, license renewals, passcodes
Sales, pricing, and general information

508-647-7000 (Phone)
508-647-7001 (Fax)

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098

For contact information about worldwide offices, see the MathWorks Web site.

Table of Contents

1. Hardware and Software requirements	7
1.1. Licenses	9
1.2. Timing information	9
1.3. UNIX/Linux installation requirements	10
2. CD-ROM content	11
3. Wizard installation	13
3.1. PolySpace Autorun	14
3.2. Choice of setup type	15
3.3. PolySpace install directories	16
3.4. PolySpace shared components	17
3.5. PolySpace Licenses	20
3.6. PolySpace Documentation	21
3.7. PolySpace Icons	21
3.8. Launching the installation wizard in batch mode	22
4. Remote Launcher installation	23
4.1. Terminology	23
4.2. Installation of the PolySpace Server	24
4.2.1. Graphical configuration of PolySpace Queue Manager	26
4.2.2. Manual configuration of the PolySpace Server	27
4.2.3. Configuration file and directory	29
4.2.4. Installing a new CPU	30
4.2.5. List of common errors	31
4.3. Installation of the PolySpace Client	32
4.3.1. The PolySpace Client configuration	33
4.3.2. Manual configuration of the Client	34
4.4. PolySpace upgrades	35

5. PolySpace Model Link SL and TL installation (only for C language)	37
5.1. MATLAB upgrade	38
5.2. MATLAB is not installed locally	38
5.3. No write access to MATLAB root	39
6. PolySpace UML Link RH installation (only for C++ language)	41
7. PolySpace C++ addin for Visual Studio (only for C++ language)	42
8. "cygwin" update on Windows	43
8.1. Updating "Cygwin" with the PolySpace Installer	44
8.2. Updating a "cygwin" version after 1.3.17	53
9. Uninstall procedure	54
9.1. Uninstalls on Windows	54
9.2. Shared components uninstall	55
9.2.1. PolySpace Viewer uninstall	55
9.2.2. Common Launcher uninstall	56
9.2.3. PolySpace C++ addin for Visual uninstall	56
9.2.4. PolySpace Model Link SL and TL uninstall	57
9.2.5. PolySpace UML Link RH uninstall	57
9.3. Un-installation on Linux/UNIX	57

Table of Figures

Figure 1: PolySpace Autorun	14
Figure 2: PolySpace choice of setup type	15
Figure 3: PolySpace shared components InstallShield.	17
Figure 4: PolySpace shared components features	18
Figure 5: PolySpace license request	20
Figure 6: Choice between Client and Server.	21
Figure 7: Client/Server architecture.	23
Figure 8: Server Configuration	24
Figure 9: Server installation process	25
Figure 10: PolySpace Remote Launcher Manager.	26
Figure 11: Server configuration files	29
Figure 12: Client Configuration	33
Figure 13: PolySpace Client configuration parameters.	34
Figure 14: PolySpace Model Link TL and PolySpace Model Link SL configuration	37
Figure 15: "cygwin" missing packages	44
Figure 16: "cygwin" setup	45
Figure 17: Choose Installation Type	46
Figure 18: Choose Installation Directory	47
Figure 19: Select Local Package Directory	48
Figure 20: Select Packages	49
Figure 21: "cygwin" setup progress	50
Figure 22: Create Icons	51
Figure 23: Installation Complete	52
Figure 24: Select components to uninstall	55

1. Hardware and Software requirements

PolySpace products requirements

Please check that the following requirements are met before installing PolySpace on a server or on a development workstation (Client usage).

	PolySpace Client	PolySpace Server
Operating System	<ul style="list-style-type: none">• On a Linux PC with RedHat Enterprise Linux Workstation 3 (32 Bits release) and 4 (32/64 Bits), Suse Linux 10.0 (32 Bits) and 10.1 (32/64 Bits).• On a Windows PC with Windows 2000 (SP4), and XP (SP2 / 32 Bits),• On a Sun Solaris 2.7 or 2.8 workstation.	
	PolySpace recommends using a Linux PC	
RAM	400 Mb minimum, 512 Mb (recommended).	1,5 Gb ECC minimum, 2 Gb ECC (recommended)
Processor	400MHz minimum, 1 GHz (recommended)	1,5 GHz minimum, 3 GHz (recommended).
Swap	Swap space should be at least twice the RAM size.	
Free hard drive space	2Gb minimum	2Gb minimum (SCSI recommended)

PolySpace plug-in requirements

Please check that the following requirements are met before installing PolySpace Model Link SL, PolySpace Model Link TL, PolySpace UML Link RH and PolySpace C++ add-in for Visual Studio.

	Release	Operating Systems
PolySpace Model Link SL		
MATLAB	R13 (SP2) and R2007a	Windows OS (see operating systems above)
Real Time Workshop Embedded Coder	4.2.1 and R2007a	
PolySpace Model Link TL		
MATLAB	R13 (SP2) and R2007a	Windows OS
TargetLink (dSPACE)	2.1 and 2.2	
PolySpace UML Link RH (Only with C++ language)		
Rhapsody (Telelogic)	7.0 to 7.3	Windows OS
PolySpace C++ add-in for Visual Studio (Only with C++ language)		
Microsoft Visual Studio.NET 2003	7.1	Windows OS
Microsoft Visual Studio 2005	8.0	

1.1. Licenses

PolySpace provides several licensing options for its products, which when combined with the supported installation types, provide a wide range of asset mobility. Asset mobility, as used here, means the ease with which product licenses can be redeployed to different users.

License Option Description:

License Option	Protection	Description
Individual	Designed computer (MAC address) or serial disk number.	You are an end user and you want to operate, install, and administer the software yourself. On a Desktop PC, an individual license is given in a text file.
Concurrent	Designed computer (MAC address) or serial disk number.	Your organization is standardized on the FLEXnet licenses manager, or you need to impose strict limits on software use for accounting or license management purposes. This kind of license needs a FLEXnet manager.

Note: see the PolySpace License guide. In particular, it describes the installation of the FLEXnet licenses manager.

1.2. Timing information

The installation procedure lasts between 5 and 10 minutes.

1.3. UNIX/Linux installation requirements

Configuration

- Ensure that the /tmp folder contains at least 10 Mb free space.

User's environment settings

- User's path needs to include <PolySpaceInstallDir>/bin.
- The PolySpace license file needs to be placed in <PolySpaceInstallCommon>/etc (see paragraph about "PolySpace install directories").
- Graphical User Interfaces of PolySpace require **CDE** Window manager on Solaris Open Software environment and **Gnome** on Linux. Behaviour on other window managers is not guaranteed.

Access right

- Read and execution access rights are required for all the PolySpace installations.



2. CD-ROM content

The CD-ROM allows installation of one or more of the available PolySpace products and features:

1. PolySpace Client for C/C++: it allows analysing and reviewing C and C++ source code.
2. PolySpace Client for Ada: it allows analysing and reviewing Ada83 and Ada95 source code.
3. PolySpace Server for C/C++: it allows analysing C and C++ source code on a dedicated server.
4. PolySpace Server for Ada: it allows analysing Ada83 and Ada95 source code on a dedicated server.
5. PolySpace Model Link SL: it allows analysing and reviewing C source code generated by The MathWorks Real-Time Workshop Embedded Coder from a Simulink® model.
6. PolySpace Model Link TL: it allows analysing and reviewing C source code generated by dSpace TargetLink code generator from a Simulink model.
7. PolySpace UML Link RH: it allows analysing and reviewing C++ source code generated by Telelogic Rhapsody.

The CD-ROM contains 4 subfolders and program files. Important subfolders and programs are described below:

1. "Setup.exe" and "setup_shared.exe" run the Graphical Install Wizard of PolySpace on a Windows platform. Shared install is dedicated to PolySpace shared components.
2. "install.sh" runs the shell script Install Wizard of PolySpace on a UNIX platform. Option `-console` enables to execute the command in a console mode. Running `install.sh -h` gives option and usages.
3. "solaris.bin" and "solaris_shared.bin" runs the graphical install wizard of PolySpace on a Solaris platform. Shared install is dedicated to PolySpace shared components.
4. "linux.bin" and "linux_shared.bin" runs the graphical install wizard of PolySpace on a Linux platform. Shared install is dedicated to PolySpace shared components.
5. "version.txt" contains the complete version number of the PolySpace product that has been burnt on the CD-ROM.
6. The "Addons" subfolder contains some useful free or under GPL license tools, available for Windows and / or UNIX platforms.
7. The "Docs" subfolder contains all the documentation associated to PolySpace products, in HTML and PDF formats. **Note that** the PolySpace licenses guide is located in this subfolder. It describes installation and available licence protection mechanisms.
8. The "Data" subfolder contains PolySpace data.
9. The "Examples" subfolder contains some results of analysis done with PolySpace tools (using PolySpace Client, Server and Model Link for each language). These results can be displayed with the PolySpace Viewer available with any of the PolySpace Clients for C/C++ and Ada.

Note: Please refer to the Getting Started document for a quick description of the PolySpace products associated to a particular language.



3. Wizard installation

Before following step by step the installation Wizard, please check that you have three mandatory materials:

- The PolySpace CD-ROM.
- A `license.dat` file accessible on the disk.
- A local user login with **administrator** rights.

After inserting the PolySpace CD-ROM on the drive, the PolySpace autorun starts. If no autorun starts after a couple of minutes, you can launch the `setup.exe` program which is located on the <CD-ROM> drive.

Then, please follow the instructions given during the graphical install. Some particular steps are detailed in the paragraphs and sections below.

The same graphical wizard is available on Solaris, Linux or Windows platforms. This section describes only a complete Windows install.

3.1. PolySpace Autorun

The installation procedure goes through steps. First one is described below:

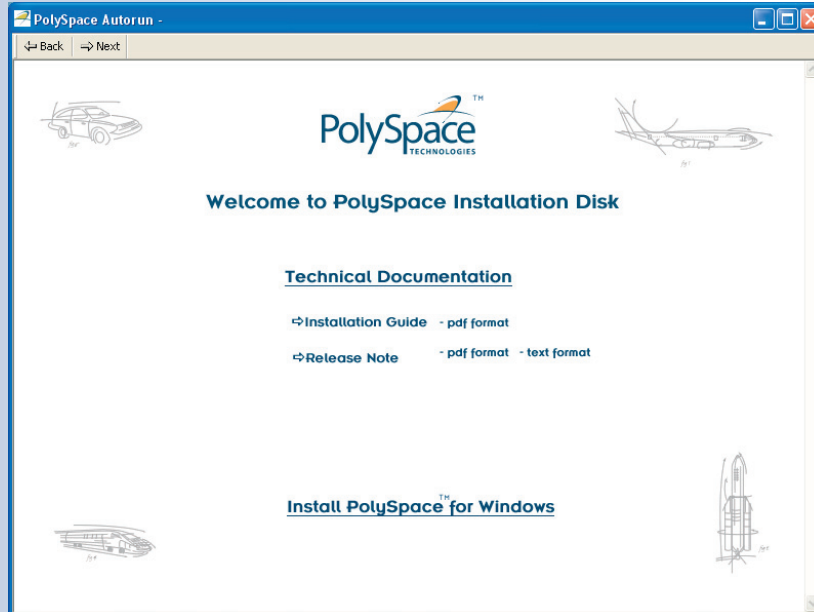


Figure 1:
PolySpace Autorun

Click on « Install PolySpace for Windows ». Then the InstallShield wizard runs, preparing the Java Virtual Machine. The installation time (progress until 100%) will be between 5 and 10 minutes depending on your computer.

3.2. Choice of setup type

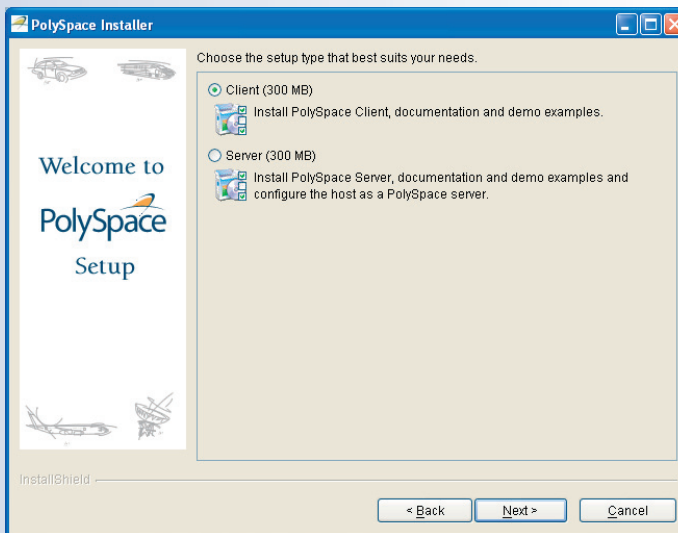


Figure 2:
PolySpace choice of setup type

Click on the adequate radio button:

- A Client install is recommended when you are on a client computer. It will allow performing a local analysis of a single file, reviewing analysis results

and sending new analysis to a dedicated server. Installation will use between 300 MB to 500 MB of disk space. A user login with local administrator rights must be used in order to install the "Cywin" module.

- A Server install is recommended when the machine is dedicated to PolySpace and will receive analysis coming from Clients. Such machines are also referred to as CPUs (see paragraph "Remote Launcher installation"). Installation will use 300 MB to 500 MB of disk space. **A local user login with local administrator rights must be used.**

3.3. PolySpace install directories

Three directories will be required in order to install PolySpace:

- The main directory contains the kernel of the installed release of PolySpace Client/Server for C/C++ and Ada. The default and proposed directory is "C:\PolySpace\PolySpaceFor<language>" where <language> can be one of the values "CandCPP" and "Ada".
- Another directory contains common components available for any language: "cygwin" library, java run time library, the PolySpace Viewer, the Remote Launcher and some add-ons available with the PolySpace products such as PolySpace C++ Visual addin and PolySpace in One Click Toolbar (also called POC). The default and proposed directory is "C:\PolySpace\PolySpace_Common".
- A third directory contains the Remote Launcher configuration file `polyspace.conf`. The directory is "C:\PolySpace\PolySpace_RLData".

In this document:

- <PolySpaceInstallDir> refers to "C:\PolySpace\PolySpaceFor<Language>" ("`/usr/local/PolySpace/PolySpaceFor<Language>`" under UNIX systems),
- <PolySpaceInstallCommon> refers to "C:\PolySpace\PolySpace_Common" ("`/usr/local/PolySpace/PolySpace_Common`" under UNIX systems).
- <PolySpaceRLData> refers to "C:\PolySpace\PolySpace_RLData" ("`/etc/PolySpace` under UNIX systems).

3.4. PolySpace shared components

During the PolySpace product installation, components used by all PolySpace products are located in the PolySpace shared area.

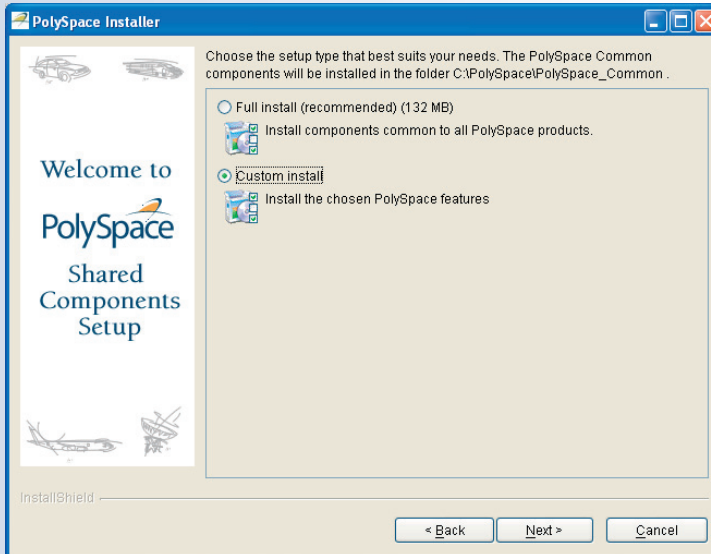


Figure 3: PolySpace shared components InstallShield.

A second InstallShield launches and proposes available components. A full installation is recommended. If you already have PolySpace installed, you may want to do a Custom install.

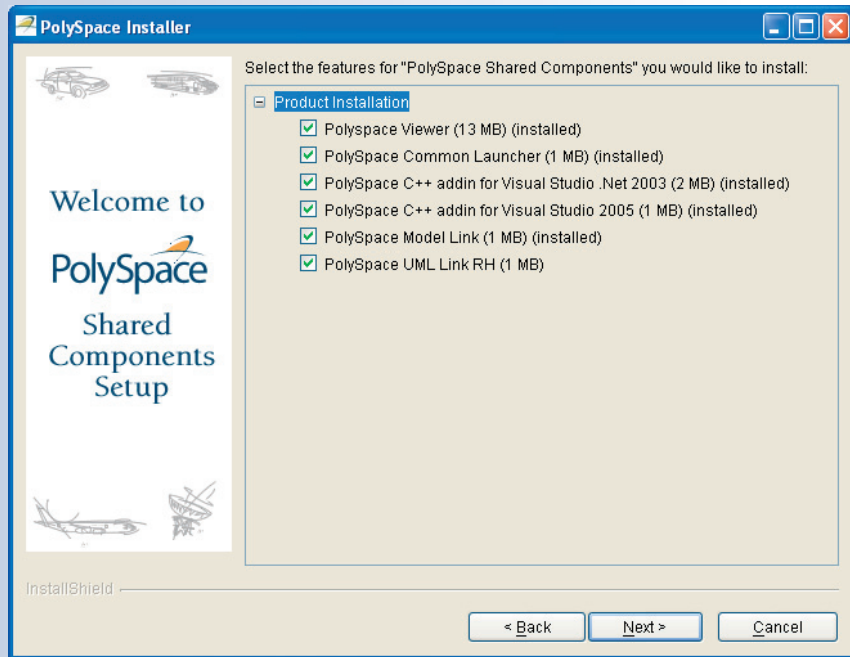


Figure 4: PolySpace shared components features

- PolySpace Viewer component

This component allows seeing the results of analyses done with PolySpace Client or Server for C/C++ and Ada in a graphical way. **This component is mandatory for any installation of a PolySpace Client.**

Note: the keyword (installed) following the component means that it is already installed.

- PolySpace Common Launcher component

This component is the graphical interface of PolySpace Launcher allowing starting any analysis. **This component is mandatory for all PolySpace Clients and Servers.**

- PolySpace C++ addin for Visual component (**Only for C++ language on Windows**)

This component allows using PolySpace tools directly from Microsoft Visual Studio. The component is proposed to the installation only if PolySpace has detected that Microsoft Visual Studio is available on the machine on which it is being installed.

- PolySpace Model Link component (**Only for C language on Windows**)

This component allows using PolySpace tools directly from within The MathWorks Simulink® associated to a particular code generator Real-Time Workshop Embedded Coder or dSpace TargetLink. The component is proposed to the installation only if PolySpace has detected that these products are available on the machine on which it is being installed.

- PolySpace UML Link RH component (**Only for C++ language on Windows**)

This component allows using PolySpace tools directly from Telelogic Rhapsody. The component is proposed to the installation only if PolySpace has detected that Telelogic Rhapsody is available on the machine on which it is being installed.

After the end of the PolySpace shared components installation, the installation of PolySpace kernel continues.

3.5. PolySpace Licenses

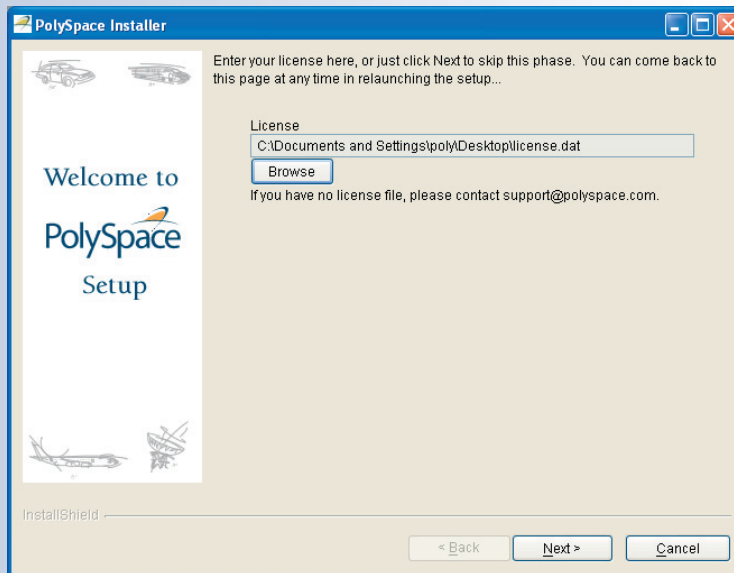


Figure 5:
PolySpace license request

Select your `license.dat` file using the **Browse** button. Please refer to the PolySpace licenses guide document for more information about available licenses and installation of the FLEXnet manager. If you do not have your license file, you can skip this step and directly click on the **Next >** button without filling in the "License" text box. You will be able to install the licence file later.

Note: On Windows, the license file location depends on the license mechanism. By default, it must be located in `<PolySpaceInstallCommon>/etc`.

3.6. PolySpace Documentation

The PolySpace Documentation requires Adobe Acrobat Reader version 5.0 or above. You can find it at www.adobe.com.

3.7. PolySpace Icons

You have now three new icons on your PC:



- PolySpace *Launcher* performs single file, multi-file or composite analysis of source code and gives the location and status of run-time errors in those files. The choice of a simple or multi-task analysis is done at launching (choice between Client Launcher or Server Launcher).

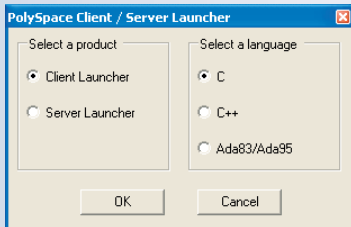


Figure 6: Choice between Client and Server.

- PolySpace *Viewer* is the graphical user interface to review the results computed by PolySpace Client and PolySpaceServer.
- PolySpace *Spooler* is the graphical user interface to manage analyses sent remotely using PolySpace Client.

3.8. Launching the installation wizard in batch mode

• Installation on a UNIX platform

To install PolySpace products under Linux or UNIX system, execute in a terminal window the following command: `<CD-ROM>/install.sh`

This script launches a textual install wizard. Then, please follow instructions described in the sections above.

Note: If you are on the **Gnome** window manager under Linux or **CDE** window manager under Solaris, it is possible to click on "Linux.bin" or "Solaris.bin" programs located at the root of the CD-ROM path in order to have access to a graphical installer.

• Silent mode

To install PolySpace products under UNIX systems in a silent mode, you can execute in a terminal window the following command:

```
<CD-ROM>/install.sh -silent -install-dir <PolySpaceInstallDir>
```

where `<PolySpaceInstallDir>` represents the install directory for PolySpace products.

Note that, by default, `-silent` mode is equivalent to a PolySpace Client installation with default mode for remote launching parameters (See "Remote Launcher installation": `auto` for server mode and `12427` for the port).

On a Windows machine, you can execute the following bat file in a "command prompt" shell:

```
<CD-ROM>\DATA\Windows\setup_silent.bat <PolySpaceInstallDir>
```

4. Remote Launcher installation

In the remote configuration mode of PolySpace, a PolySpace Server can receive PolySpace analysis requests from several PolySpace Clients.

4.1. Terminology

- The PolySpace Clients are computers asking for PolySpace analyses to be performed on CPUs.
- The PolySpace Servers - which only analyze source code coming from Clients - are named CPUs.
- The PolySpace Queue Manager (QM) manages the analyses coming from the Clients.
The machine on which it is installed can also be a Server but doesn't have to.

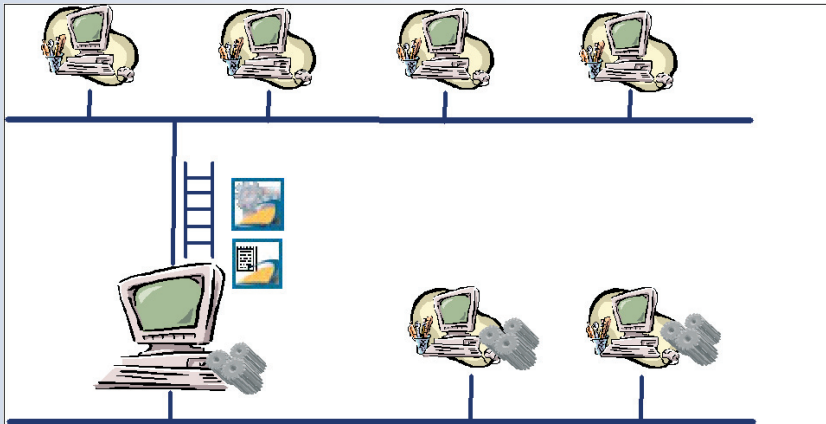


Figure 7: Client/Server architecture.

4.2. Installation of the PolySpace Server

Please remember that it is mandatory to install PolySpace Server and the Remote Launcher using a **local** user login with **administrator** rights.

After having chosen the **Server** radio button (See Figure 2), you shall define the Server parameters:

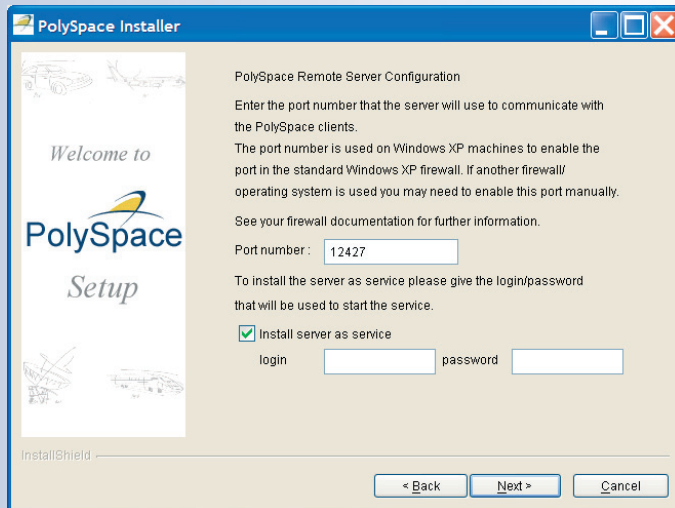


Figure 8: Server Configuration

- **Server Configuration:** You can choose the port number through which the communication with Clients will be established. It defaults to 12427. You will have to use the same port number when you will be installing PolySpace Clients.
- To install a server as a service on a Windows machine, click on "Install server as service" and type in the login and password used to start the server as a Windows service.

Note: The layout of this screen is for a Windows installation.

- To install a server as a daemon on a Linux or Solaris machine, click on "Install server as service" and type in login used to start server as a daemon (no password is required).

Note: if the radio button has not been ticked, you will have to launch the Queue Manager manually (See next paragraph "Manual configuration of the Server").

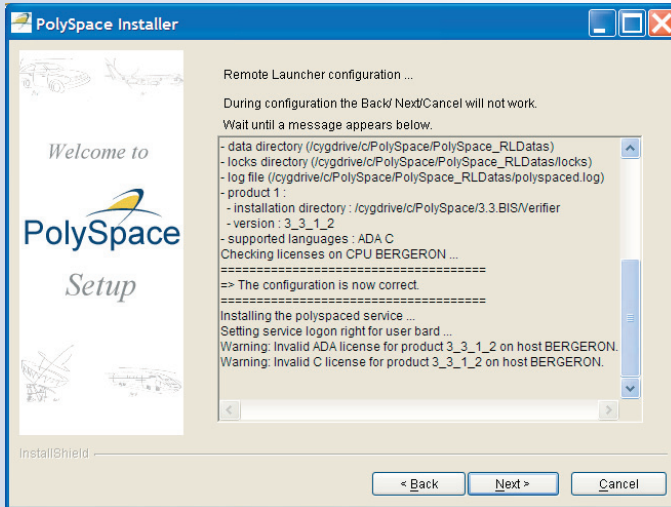


Figure 9: Server installation process

• **Server Installation process:** The configuration and installation of the Server for the Remote Launcher is finished when the message "The configuration is now correct" is displayed. Clicking on the button will finish the PolySpace installation.

By default, the Server you just installed becomes the first CPU of the architecture. All the information listed in the dialog box is saved in the `<PolySpaceInstallCommon>/../PolySpace_RLData/polyspaced.log` file (`/var/PolySpace/polyspaced.log` file under UNIX). It contains information about product paths checking, licenses, creation and start of the service (daemon).

If a problem occurs, a message appears at the bottom of the screen or in the **log file** `polyspaced.log` (see location in "Configuration file and directory"). Most problems are related to right accesses and duplicate or bad information associated to a key (see "List of common errors").

4.2.1. Graphical configuration of PolySpace Queue Manager

The Queue Manager can be configured or updated graphically through the "Remote Launcher Manager" which is accessible in the Windows start menu at "Start>Programs>PolySpace". This launches <PolySpaceInstallCommon>/RemoteLauncher/PSTRManager.exe and updates the polyspace.conf file.

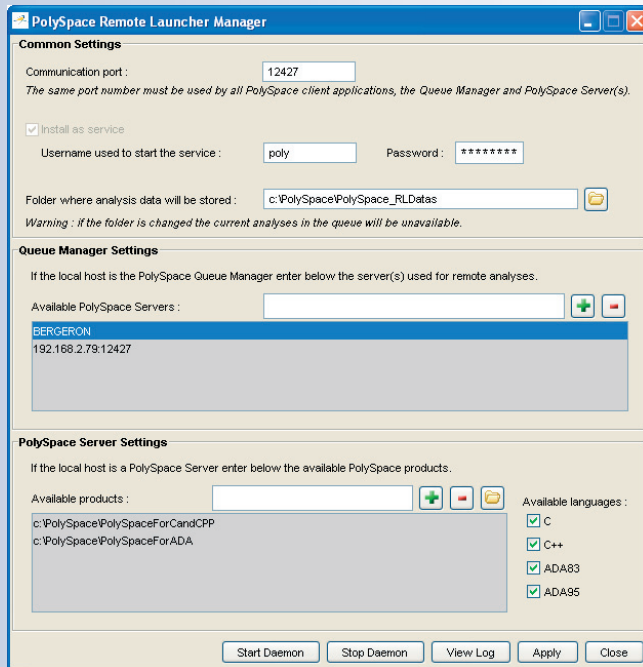


Figure 10:
PolySpace Remote Launcher Manager.

The "Common Settings" part allows setting port number and locations of the Queue Manager as a Windows service. Note that it is mandatory to stop the current Queue Manager before any change or update of the parameters. Please use first the "Stop Daemon" button.

The "Queue Manager Settings" part allows setting the IP address of the Queue Manager and all attached CPUs. The syntax is "<describing name/IP address>[:<port number>]".

In the figure above, "BERGERON" is the Queue Manager and "192.168.2.79:12427" represents a CPU with a specific port number.

The "PolySpace Server Settings" part allows setting PolySpace products locations and available languages.

Note that all information displayed during the execution of the "Start Daemon" and "Stop Daemon" commands are also written in the `polyspaced.log` file in `<PolySpaceInstallCommon>/../PolySpace_RLData/` (`/var/PolySpace` under UNIX).

4.2.2. Manual configuration of the PolySpace Server

A batch command `<PolySpaceInstallCommon>/RemoteLauncher/bin/setup-remote-launcher` can be used to create or destroy the `polyspace.conf` configuration file.

The `polyspace.conf` file may be found:

- On a Windows system at: `<PolySpaceInstallCommon>/../PolySpace_RLData/`.
- On a UNIX system at: `/etc/PolySpace/`.

The configuration file contains fields which set a Queue Manager and CPU architecture:

- `product=.` determines the installation directories of PolySpace Verifier. It is possible to have one per language.
- `lang=.` determines the supported analysis languages. The language variable specifies the supported language declaration (CPU only). The values may be: C, C++, CPP, Ada, Ada83, ADA, Ada95 and ADA95. More than one language can be specified.
- `username=.` specifies the user name used to run the daemon process. If the daemon is started as `root` on UNIX, it will automatically change its user ID to run as the given user.
- `datadir=.` identifies where the data sent remotely are stored.

- `locksdir=.` identifies where the locks are stored. `<datadir>/locks` will be used by default.
 - `port=.` corresponds to the TCP port the daemon is listening to (default value is **12427**). It can be specified in the range *10000* to *15000*. It depends on the available OS requirements.
- Note:** The port number associated to the Queue Manager has to be open in the server firewall.
- `cpu=.` the CPU name and port as specified for Queue Manager and CPU. Syntax can be `<address IP or name of CPU>[:port number]`. If more than one `cpu=` is specified, others are CPUs to which the Queue Manager can dispatch analysis.

Note: if this field is missing in the configuration file, the machine will only be a CPU.

For any change in the configuration file, except for the `cpu=` field, the associated daemon/service manager need to be stopped and re started:

- Use `<PolySpaceInstallCommon>/RemoteLauncher/bin/setup-remote-launcher -stop-service` to stop the service. Note that any standard program on OS can stop the service or daemon.
 - Use `<PolySpaceInstallCommon>/RemoteLauncher/bin/setup-remote-launcher -start-service` to start or restart the service. Note that any standard program associated to the OS can restart the service or daemon.
- Note:** `<PolySpaceInstallCommon>/RemoteLauncher/bin/setup-remote-launcher -h` gives available options.

If the Queue Manager has not been configured as a service (daemon) during installation, the following command allows launching and stopping the Queue Manager manually:

- `<PolySpaceInstallCommon>/RemoteLauncher/bin/InstallPolySpaceService.bat` will install the Queue manager service on a Windows platform.

Note: The Windows service can also be managed like other standard services

(by using, for example, the "My Computer>Manage>Services and Applications" menu on a Windows XP machine).

- `<PolySpaceInstallCommon>/RemoteLauncher/bin/setup-remote-launcher -install-service` will install the Queue manager daemon on a UNIX platform.

Note: the PolySpace daemon can also be managed like other daemon in the `/etc/init.d` install services directory.

`polyspaced` is the service/daemon stopped/launched using the following commands:

- Use `<PolySpaceInstallCommon>/RemoteLauncher/bin/polyspaced -stop` to stop the Queue Manager.
- Use `<PolySpaceInstallCommon>/RemoteLauncher/bin/polyspaced -start` or restart the Queue Manager.

Note: `<PolySpaceInstallCommon>/RemoteLauncher/bin/polyspaced -h` gives available options.

4.2.3. Configuration file and directory

In the <PolySpaceInstallCommon>/../PolySpace_RLData directory you can find the following files and directories:

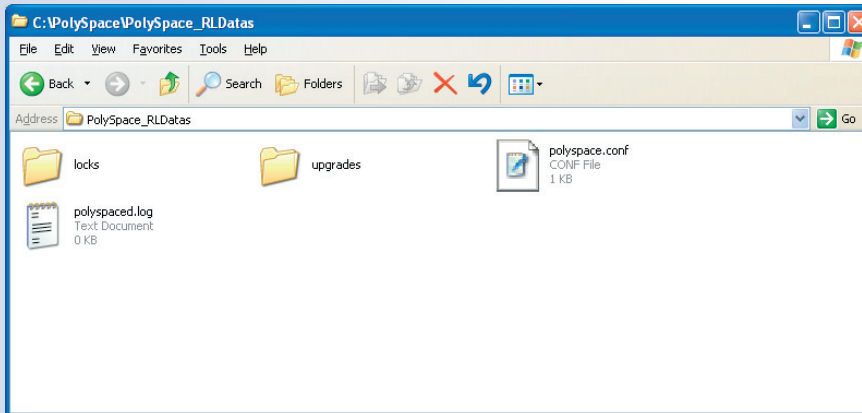


Figure 11: Server configuration files

- **polyspaced.log** contains installation log information and all communications between Clients and Queue Manager during the Queue Manager life.
- **queue_infos.txt** contains information about the analyses sent to the Queue Manager. These information are those seen through the Spooler dialog box.

- The `locks` directory contains all the locks file for each analysis.
- `polyspace.conf` is the configuration file. Below is an example of `polyspace.conf` :

```
product=c:\PolySpace\3.3
lang=CPP
lang=C
datadir=c:\PolySpace\PolySpace_RLDatas
username=bard
cpu=BERGERON
cpu=192.168.2.79:12428
```

In this example, the BERGERON CPU has been configured to be a Queue Manager and CPU for C and C++ analyses on default port 12427. The product installed in `c:\PolySpace\3.3` will be used. Another CPU is available at the IP address `192.168.2.79` with a specific port number (12428).

On a UNIX system, the directory location containing locks and `datadir` is, by default, `/var/PolySpace`.

4.2.4. Installing a new CPU

Installing a new CPU is very simple: just follow the standard PolySpace Server installation. Then, edit the `polyspace.conf` of the CPU which has been chosen as Queue Manager, and add the new CPU using the field `cpu=<describing name/IP address>[:<port number>]` (see example above where there are two CPUs).

Note: on the new CPU, it is important to remove the field `cpu=` in the `polyspace.conf` file to avoid having this machine being also seen as a "Queue Manager".

4.2.5. List of common errors

Here is the list of error messages related to the `polyspace.conf` file:

- "Error: The data directory path should not contain white space.";
- "Error: At least one language should be supported by the CPU.";
- "<polyspace.conf>:<line_number>: Duplicate entry for product <version value>.";
- "<polyspace.conf>:<line_number>: Unknown language : <value>.";
- "<polyspace.conf>:<line_number>: Duplicate entry for language <value>.";
- "<polyspace.conf>:<line_number>: Duplicate entry for <queuemanager name>.";
- "<polyspace.conf>:<line_number>: Duplicate entry for <value>.";
- "<polyspace.conf>:<line_number>: Duplicate entry for 'username'.";
- "<polyspace.conf>:<line_number>: Duplicate entry for 'port'.";
- "<polyspace.conf>:<line_number>: Invalid port number : <value>";
- "<polyspace.conf>:<line_number>: Invalid CPU name.";
- "<polyspace.conf>:<line_number>: Duplicate entry for CPU <hostname> ";
- "<polyspace.conf>:<line_number>: Unknown key : <value> ";
- "<polyspace.conf>:<line_number>: Parse error.";
- "No PolySpace product found for language <value>.";
- "Warning: Invalid <lang> license for product <version> on host <hostname> ";
- "The PolySpace product <version> was not found.";

When getting any of these errors, please stop the `polyspaced` service (daemon) and update the `polyspace.conf` file. Then restart the service (daemon).

Here is the list of errors related to access rights problems:

- "Fatal error: Directory <dir> is not accessible.";
- "Fatal error: Could not change the owner of <path> to <username> ";
- "Fatal error: Could not open file <file name> for reading.";
- "Fatal error: Could not set <permission> permissions on <path> "
- "Fatal error: Could not open file <file name> "

When getting these errors, please check access rights of the username declared in the `polyspace.conf` file in the directory <dir>, <file name> or <path> (given in the error message). Also check whether you didn't change something linked to the "polyspaced" service (daemon) without stopping it. Other errors may be related to OS system, network and unavailability of standard commands (such as "tar", "gzip", etc.).

4.3. Installation of the PolySpace Client

In the remote configuration mode of PolySpace, a PolySpace Client can send PolySpace analyses as soon as the compilation phase is finished. Clients are computers asking for PolySpace analyses to be performed on CPUs. Each Client, via the PolySpace Spooler, manages its own analyses and can download completed results.

4.3.1. The PolySpace Client configuration

After having chosen the **Client** radio button (see Figure 2), you shall define the port number.

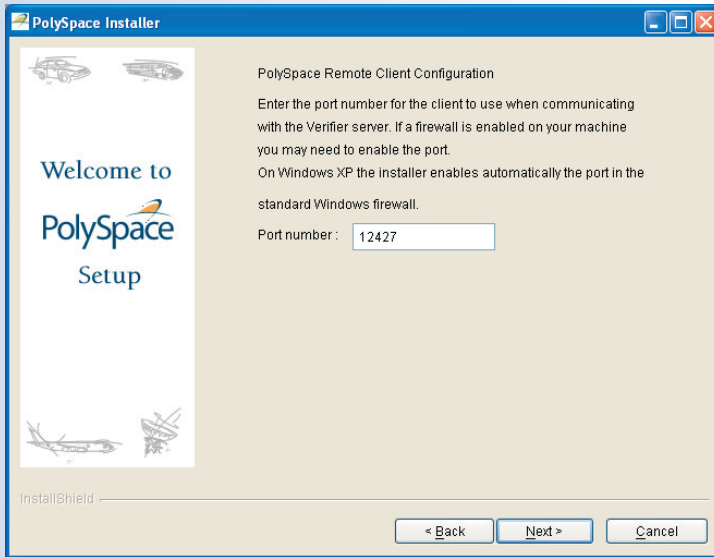


Figure 12:
Client Configuration

The port number allows the communication with the PolySpace Server to be established. The port number by default is 12427 and needs to be the same as the port chosen when you installed the PolySpace Server. The procedure is identical for a UNIX PolySpace Client machine.

4.3.2. Manual configuration of the Client

By default, the PolySpace Client is in "auto" mode. It means that the Remote installation of PolySpace tries to find an available PolySpace Server configured as a Queue Manager on the network.

If it finds more than one PolySpace Server, an error dialog box will be displayed with the name of all servers found.

Note carefully all names and acknowledge the error dialog box.

If none is available, an error message will be displayed when launching an analysis or when executing any PolySpace command. The error message is the following: "No QueueManager server found".

It is possible to specify a Queue Manager server associated to a specific port number.

To do so you must open



the associated graphical user interface (`<PolySpaceInstallDir>/bin/polyspace-launcher` on UNIX) and set the "Remote Launcher Preferences" available through the menu "Edit>Preferences>Remote Launcher":

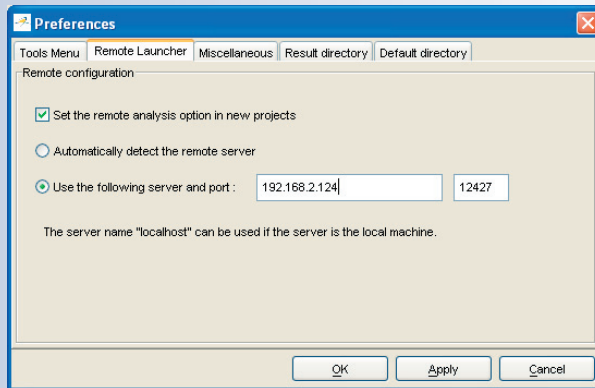


Figure 13: PolySpace Client configuration parameters.

By default, the first radio button is ticked.

Click on the second radio button and enter an IP address (recommended for network performances) or a server name associated to the port number that has been chosen during "Server" installation.

These preferences are stored in the associated preferences file in `<user home directory>/ .rte` directory (`c:\Documents and Settings\<user name>\ .rte` on a Windows system):

the `PolySpace-Launcher.prf` file contains the preferences associated with a PolySpace analysis.

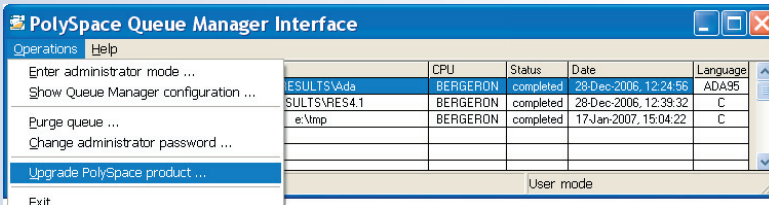
Note: Even if the file can be modified by a ASCII editor, it is recommended to use the graphical user interface.

4.4. PolySpace upgrades

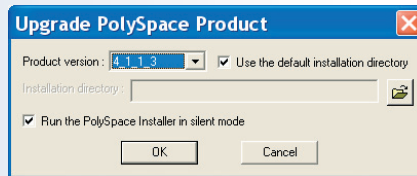
• A PolySpace Client

When the remote launcher has been installed, The Queue Manager interface (or PolySpace Spooler) is available to manage analyses sent by a PolySpace Client.

The Menu "Operations>Upgrade PolySpace Product ..." allows the upgrade of a PolySpace Client.



If, on the Server side, in the <PolySpaceInstallCommon>/../PolySpace_RLData/upgrades directory, a new release of PolySpace is available with the format: <version>.tar or <version>.tar.gz, it is possible to upgrade the PolySpace client:



Note that the tar file has to contain the setup.exe file at its root.

After having chosen options (installation directory and silent modes), the "PolySpace Queue Manager Interface" downloads the complete setup package (approximately 600 Mb) and unpacks it. This can take several minutes, depending on network speed and system performances. Then, a standard installation begins. Note that it is recommended to close all PolySpace products before launching an installation.



• A new release for a Server or CPU

When you are installing a new product on a Server you need to check whether the PolySpace Server daemon is active or not.

◆ On an active Server, just follow a standard PolySpace **Client** installation.

After the installation, you will have to edit the `polyspace.conf` file in

`<PolySpaceInstallCommon>/../PolySpace_RLData` (for a Windows OS)

or in `/etc/PolySpace` (for a UNIX OS) and add the two following lines:

- `product=<path to the new product>`
- `lang=<Language>`

◆ On an inactive Server, you can choose the PolySpace **Server** installation. The field "product=" will be filled in automatically. You will still have to add the `lang=` parameter if you are installing PolySpace for a language you didn't have already.

Do not forget to update the `polyspace.conf` of the PolySpace Server which serves as a Queue Manager, by adding a new entry `cpu=`.

Note: it is possible to install different versions of PolySpace for a given language. The PolySpace Server will automatically send analysis to the right version.

5. PolySpace Model Link SL and TL installation (only for C language)

The PolySpace Model Link SL and PolySpace Model Link TL provide automatic error detection for code generated from The MathWorks Simulink® models. Both products consists in two components: a Simulink® PolySpace library with associated blocks and a "Back to model" (HTML links in the code source) extension in the PolySpace Viewer that allows direct navigation from a run-time error in the auto-generated code to the corresponding block in the Simulink® model.

The PolySpace installer automatically detects that MATLAB/Simulink components are installed and displays the following window:

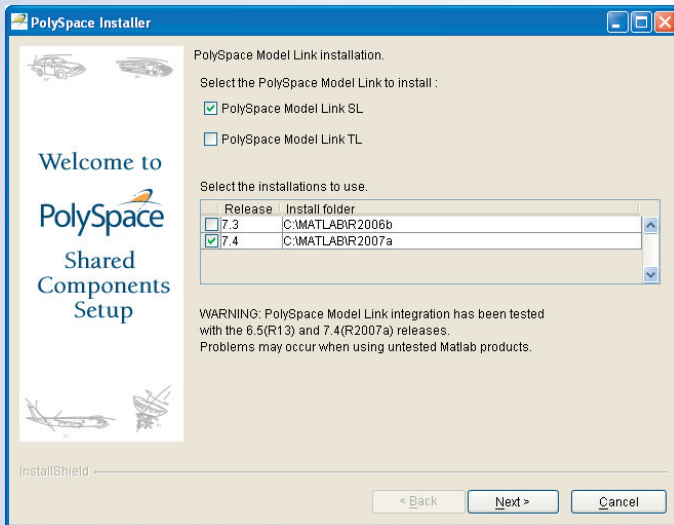


Figure 14:
PolySpace Model Link TL
and PolySpace Model Link SL
configuration

You just have to select one or more code generators and a PolySpace sub-directory is copied into each available version of MATLAB selected. The original content of the PolySpace sub-directory can be found in the <PolySpaceInstallCommon> \PolySpaceModelLink directory.

5.1. MATLAB upgrade

If you upgrade your MATLAB release, you have to re-install the PolySpace Model Link TL or the PolySpace Model Link SL for that release. If you don't have the PolySpace CD, please follow the instructions below:

- Copy the existing `<<matlabroot (old)>>\polyspace` folder into your new MATLAB release `<<matlabroot (new)>>`
- Copy the existing `<<matlabroot (old)>>\toolbox\local` into your new MATLAB release `<<matlabroot (new)>>\toolbox\local`
- Edit the new `matlabrc` file. Type "edit matlabrc" in the MATLAB command window, go to the end of the file, and add the following lines:
%% POLYSPACE CONFIGURATION
polyspacerc;
%% END OF POLYSPACE CONFIGURATION
- Run MATLAB once in the "regserver" mode, so as to post-install your MATLAB release on the local machine. To do so, run the following command line: `C:\<<matlabroot (new)>>\bin\matlab.bat /regserver /r quit`. Make sure you replace `<<matlabroot (new)>>` with the real path.

Note: When PolySpace is already installed, it is possible to install only the PolySpace Model Link TL or PolySpace Model Link SL by using again the standard PolySpace Installer of the shared components on the CD-ROM and choose the "Custom Install" option.

5.2. MATLAB is not installed locally

This would mean that the registry entries would not contain any reference to MATLAB. Thus, PolySpace Installer would not detect it and would skip the installation of the PolySpace Model Link tools.

To workaroud it, you have to do a manual installation as follows:

- Insert the PolySpace CD in your CD-ROM drive
- Locate the corresponding windows drive (hereafter 'D:\') and copy the entire folder "D:\Data\Windows\Addin\Mathworks\polyspace" into `<<matlabroot>>`
- Copy the file `D:\Data\Windows\Addin\Mathworks\startup\polyspacerc` into `<<matlabroot>>\toolbox\local`

- Edit `<<matlabroot>>\toolbox\local\matlabrc` and add the following lines at the end of the file

```
%% POLYSYSPACE CONFIGURATION
polyspacerc;
%% END OF POLYSYSPACE CONFIGURATION
```
- Run MATLAB once in the "regserver" mode, so as to post-install your MATLAB release on the local machine. To do so, run the following command line: `C:\<<matlabroot>>\bin\matlab.bat /regserver /r quit`. Make sure you replace `<<matlabroot>>` with the real path.

5.3. No write access to MATLAB root

PolySpace Model Link SL or TL does not support an installation outside `<<matlabroot>>`.

However, if you wish to install the MATLAB files in another place, you'll need to perform the following actions:

- Create a Matlab environment variable or m-function call named 'polyroot', which will return (for instance) `C:\my_folder\PolySpace\release_X_Y`
- Insert the PolySpace CD in your CD drive
- Locate the corresponding windows drive (hereafter 'D:\') and copy the entire folder `"D:\Data\Windows\Addin\Mathworks\polyspace"` into `<<polyroot>>`
- Copy the file `D:\Data\Windows\Addin\Mathworks\startup\polyspacerc` into `<<polyroot>>`
- Edit the files normally copied into the `<<polyroot>>\polyspace` folder
- Replace `matlabroot` by `polyroot` in the following m-functions at the lines indicated in the following table.

File name and Path

PolySpaceConfig.m
 PolySpaceFor-Beacon.m
 PolySpaceFor-Beacon.m
 PolySpaceFor-Beacon.m
 PolySpaceFor-Beacon.m
 PolySpaceFor-EmbeddedCoder.m

 PolySpaceFor-EmbeddedCoder.m
 PolySpaceFor-EmbeddedCoder.m
 PolySpaceFor-EmbeddedCoder.m
 PolySpaceFor-TargetLink.m
 PolySpaceFor-TargetLink.m
 PolySpaceFor-TargetLink.m
 1193
 psmenus/ polyspace_read_-icon_logo.m 3
 psmenus/ polyspace_read_-logo.m 2

Line Details of what to change

```

90  template_cfg = fullfile(matlabroot, 'polyspace', 'cfg', codegen);
115 template_cfg = fullfile(matlabroot,'polyspace','cfg','templateBeacon.cfg');
497 origstubfile = fullfile(matlabroot, 'polyspace','stub','__pststubs_be.c');
499 origppcmdfile = fullfile(matlabroot, 'polyspace','stub','ppcom_be.sh');
515 i1 = sprintf('%s', fullfile(matlabroot, 'polyspace', 'include'));
172 template_cfg = fullfile(matlabroot,
    'polyspace','cfg','templateRTWEmbeddedCoder.cfg');
610 origstubfile = fullfile(matlabroot, 'polyspace','stub','__pststubs_ec.c');
612 origppcmdfile = fullfile(matlabroot, 'polyspace','stub','ppcom_ec.sh');
632 i1 = sprintf('%s', fullfile(matlabroot, 'polyspace', 'include'));
162 template_cfg = fullfile(matlabroot, 'polyspace','cfg','templateTargetLink.cfg');
592 origppcmdfile = fullfile(matlabroot, 'polyspace', 'stub', 'ppcom_tl.sh');
  

    i1 = sprintf('%s', fullfile(matlabroot, 'polyspace', 'include'));
3  logo = fullfile(matlabroot,'polyspace','icons','icon_logo_small.gif');
2  logo = fullfile(matlabroot,'polyspace','icons','ps_logo.gif');
  
```

Note: if you wish to automate these changes: you can replace:

- all occurrences of "matlabroot, 'polyspace'" by "polyroot, 'polyspace'"
- all occurrences of "matlabroot, 'polyspace'" by "polyroot, 'polyspace'"

Then, the script "polyspacerc" must be run when opening MATLAB to enable the user to run PolySpace tools. This can be done as follows:

- Duplicate the MATLAB icon, and edit its properties: change the "Start in" text box into <<polyroot>>
- Edit the startup.m file located under "polyroot" (or create that file if it does not exist), and add "polyspacerc" on the first line

6. PolySpace UML Link RH installation (only for C++ language)

PolySpace UML Link RH provides an addin inside Telelogic Rhapsody. It consists in two components: a PolySpace library and a "Back to model" extension in the PolySpace Viewer that allows direct navigation from a run-time error in the auto-generated code to the corresponding block in the UML model.

The PolySpace Installer automatically detects that Rhapsody components are installed and installs PolySpace UML Link RH if selected in the "Custom Install" window or during a "Full Install" of the "Shared components".

If Rhapsody is installed after a PolySpace Client/Server for C/C++ product, you need to go through the standard PolySpace installation to install PolySpace UML Link RH.

By default, a `<PolySpaceInstallCommon>/PolySpaceUMLLink` directory is created. It contains all the components of PolySpace UML Link RH. The PolySpace installer also modifies two files associated with Rhapsody:

1. A line is added to the `rhapsody.ini` file located at the root of C: or in C:\WINNT:
CopyVBA=<PolySpaceInstallCommon>\PolySpaceUMLLink\bin\PolySpace.vba
where `<PolySpaceInstallCommon>` represents the path to the `PolySpace_Common` directory.
2. Some lines are added to the `<RhapsodyInstallDir>\Share\Properties\site.prp` file as follows:
Subject General
Metaclass Model
Property HelpersFile String
"<PolySpaceInstallCommon>\PolySpaceUMLLink\etc\PolySpace.hep"
End



7. PolySpace C++ addin for Visual Studio (only for C++ language)

PolySpace Installer automatically detects that Visual Studio is available and installs the addin if selected in the "Custom Install" window or during a "Full Install" of the "Shared components".

When a PolySpace Client for C/C++ is already installed, you can install again the addin in batch using the following command: `<PolySpaceInstallCommon>/VisualInterface/VisualInstall.bat`



8. "cygwin" update on Windows

On Windows, PolySpace uses Cygwin as an emulator of UNIX. Please note that PolySpace Viewer does not require Cygwin.

By default, PolySpace installs Cygwin in `<PolySpaceInstallCommon>\PST_Cygwin`.

If "cygwin" has already been installed on the machine, the PolySpace installer will check the release number of all "cygwin" modules.

If the installed "cygwin" release number is prior to **1.3.17**, if a "cygwin" module required by PolySpace is missing or if the version of any module required by PolySpace is prior to the release number needed to install PolySpace, the PolySpace installation procedure will stop and will display a **"cygwin" installation problem**.

- ▶ If the installed "cygwin" release number is higher than 1.3.17 and some of the packages required by PolySpace are missing, please go to "Updating a "cygwin" version after 1.3.17" section of this document.
- ▶ If you see the error message but are not in the situation above, you will have to update the installed "cygwin" version with the PolySpace Installer. Please read the next section of this document.

To know your "cygwin" version number, please use the `"cygcheck -s"` command.

The result of this command contains the version number of `"cygwin1.dll"` - which is actually the version number of your "cygwin" - as well as the list of all "cygwin" modules installed along with their version number. The complete list of modules required by PolySpace and their release numbers is available at the end of this document.

8.1. Updating "Cygwin" with the PolySpace Installer

Please follow the 9 steps below.

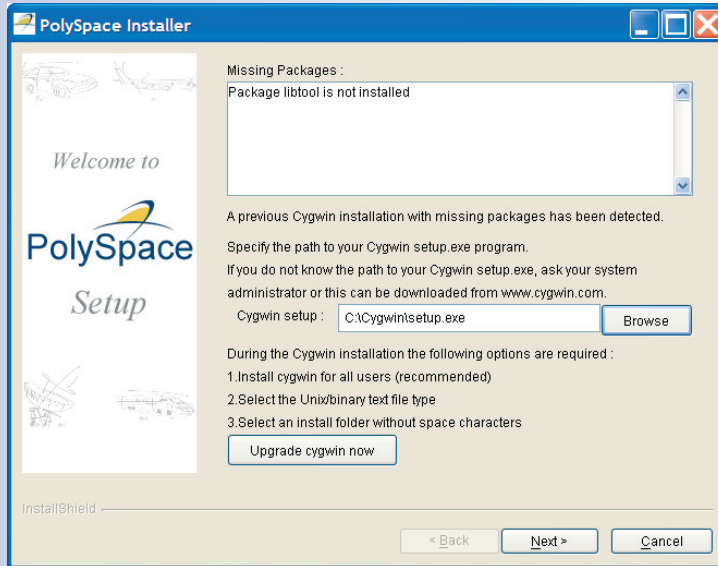


Figure 15:
"cygwin" missing packages

- **Cygwin update Step 1/9:** On the first screen you see the list of missing cygwin packages that PolySpace requires. The following steps are **mandatory before** clicking on :
- ◆ **Cygwin update Step 1'/9:** You need to select the right Cygwin setup which will allow you to install missing packages. On the figure above, the setup.exe command has been chosen using
- ◆ **Cygwin update Step 1"/9:** Click on

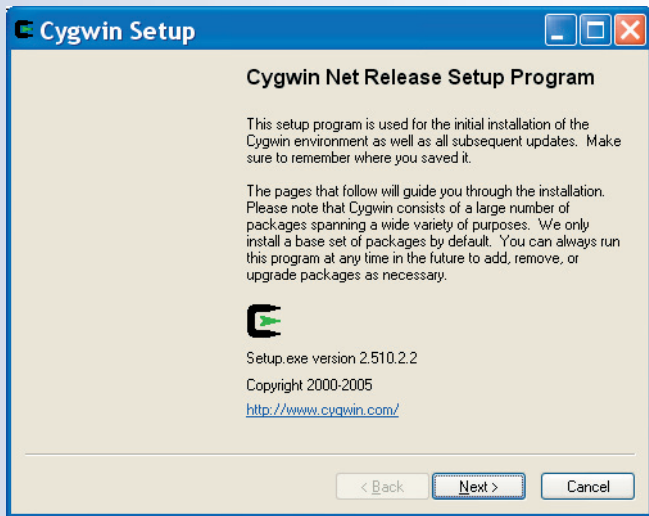


Figure 16: "cygwin" setup

- **Cygwin update Step 2/9:** Click on

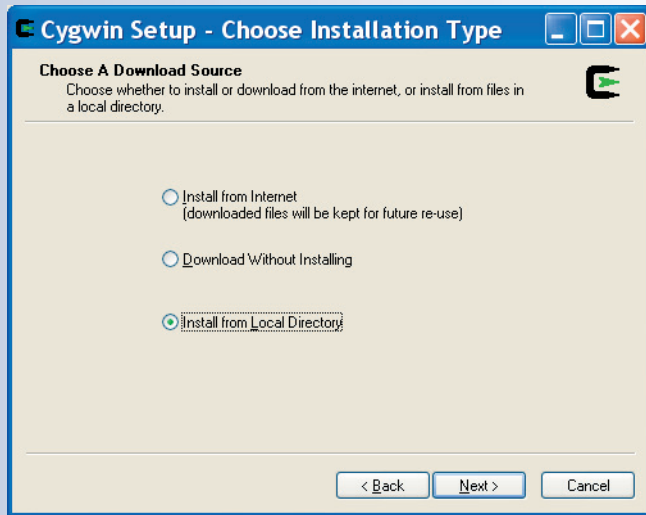


Figure 17: Choose Installation Type

- **Cygwin update Step 3/9:** Select where the Cygwin packages have been stored. On the figure above, the radio button "*Install from Local Directory*" has been chosen because packages are located on a local disk. Then click on

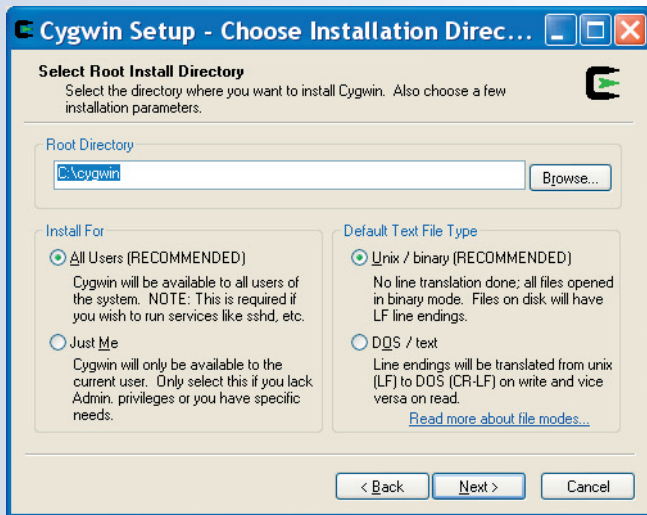


Figure 18: Choose Installation Directory

- **Cygwin update Step 4/9:** Click on and select the directory where "Cygwin" has been installed as the *Root Directory* (for example "C:\Cygwin"). Select "All Users" in the *Install For* section and "UNIX" in the *Default Text File Type* section. Then, Click on

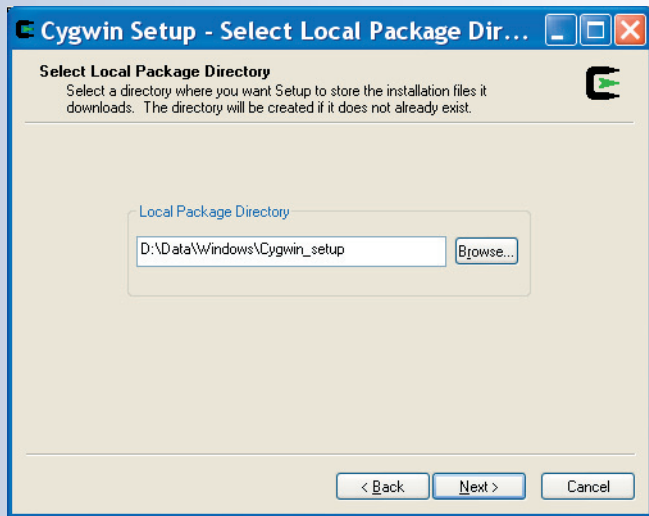


Figure 19: Select Local Package Directory

- **Cygwin update Step 5/9:** Click on and select the directory where we can find the missing packages in the *Local Package Directory* section. Then click on

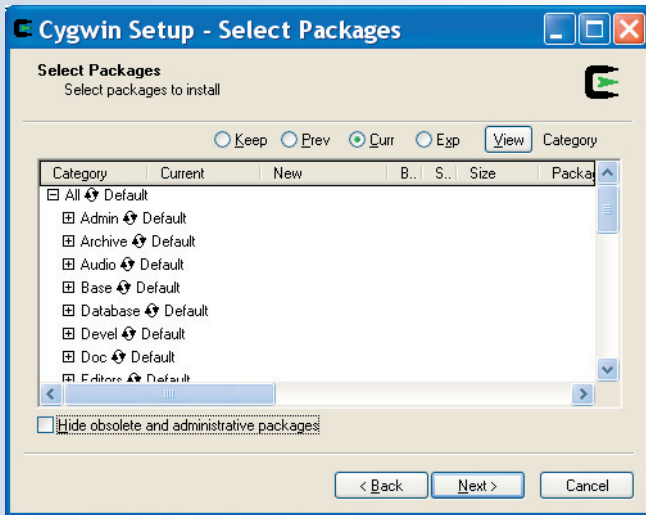


Figure 20: Select Packages

- **Cygwin update Step 6/9:** Expanding the right category you will add missing packages that PolySpace Installer required during Step 1. Then click on

Note that sometimes, some of the missing packages may be found in the *"obsolete and administrative packages"* sub-category.

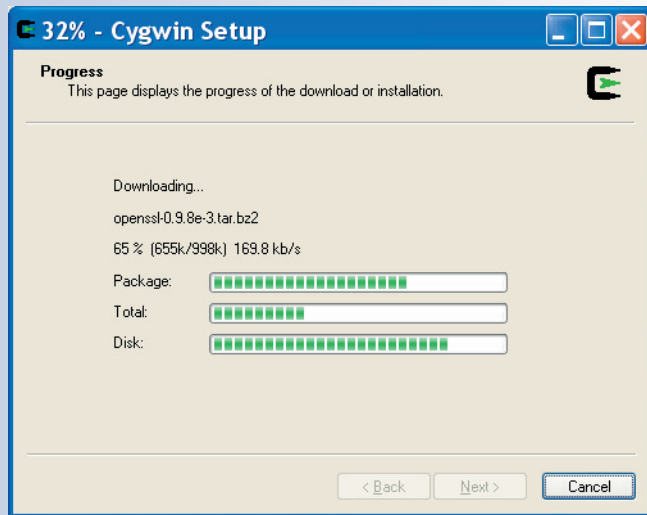


Figure 21: "cygwin" setup progress

- **Cygwin update Step 7/9:** The installation lasts between 3 and 5 minutes.

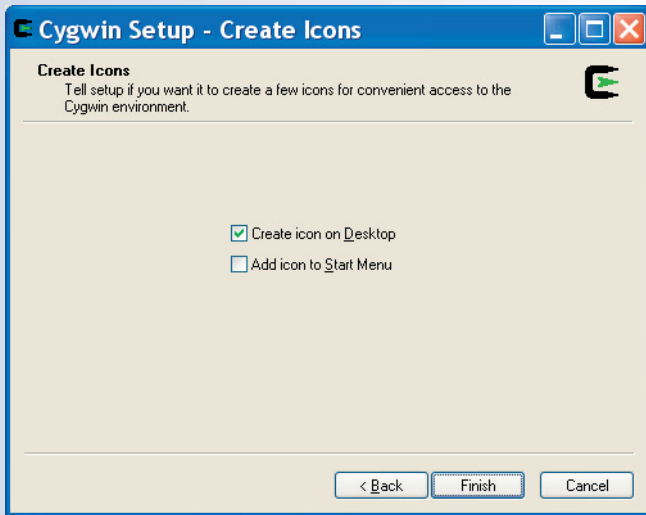


Figure 22: Create Icons

- **Cygwin update Step 8/9:** Select "Create icon on Desktop" and then, click on

Finish

Note: A DOS window appears and disappears for a few seconds,.

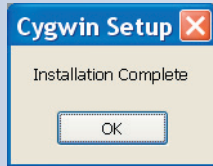


Figure 23: Installation Complete

- **Cygwin update Step 9/9:** The installation is finished. Click on PolySpace installation can now continue. Please go back to "Cygwin update step 1/9" and click on the button.

At the end of the installation, if another message indicates a missing package, please refer to the "PolySpace_Install" log file in the <PolySpaceInstallDir> directory and start back at **Step 1/9**. In that case, please follow again the steps and select the packages according to the message in the "PolySpace_Install.log" file..

You will find below an example of installation message that you may read in the *PolySpace_Install.log* file. Here, the package named "binutils" needs to be updated.

```
(Jan 24, 2005 9:10:16 AM), Setup.product.install, polyspace.StopIfBadcygwin err, Expected version 20021117-1 of package binutils but found version 20021107-2
```

8.2. Updating a "cygwin" version after 1.3.17

If you already have in the "C:\cygwin" directory a more recent version of "cygwin" than the one that is included in the PolySpace installation, the missing packages will not be available on the PolySpace CD-ROM. In that case, please download the "cygwin" installation files corresponding to your current "cygwin" installation from <http://www.cygwin.com> and install all missing packages. Packages required by PolySpace are listed below.

List of modules (package name and version) required by PolySpace:

- Ash 20020731-1
- base-files 1.1-1
- base-passwd 1.0-1
- binutils 20021117-1
- cygutils 1.1.3-1
- "cygwin" 1.3.17-1
- diffutils 2.8.1-1
- file 3.37-1
- fileutils 4.1-1
- grep 2.5-1
- gzip 1.3.3-4
- indent 2.2.8-1
- initscripts 0.9-1
- less 378-1
- libkpathsea3 1
- libtool 1
- make 3.79.1-7
- mktemp 1.4-1
- sed 4.0.1-1
- sh-utils 2.0-3
- tar 1.13.25-1
- unzip 5.50-1
- w32api 2.1-1
- zip2.3-2

Note: "cygwin" release after **1.5.x** does not offer any more fileutils, libtool and sh-utils packages. Nevertheless, you can find them in the "obsolete and administrative packages" part of the "Select packages" install interface.



9. Uninstall procedure

The same graphical wizard is available on Solaris, Windows or Linux platforms. This section only describes a complete Windows uninstall.

9.1. Uninstalls on Windows

This uninstall of PolySpace can be performed through the standard interface available on a Windows machine by clicking on "Start> Settings>Control Panel > Add or Remove Programs".

The un-installation procedure goes through several steps. Please follow the instructions and see the paragraphs below for specifics components, if necessary.

At the end of uninstall, the "<PolySpaceInstallDir>" and "<PolySpaceInstallCommon>" folder and subfolders may remain after the reboot. They can be removed manually.

9.2. Shared components uninstall

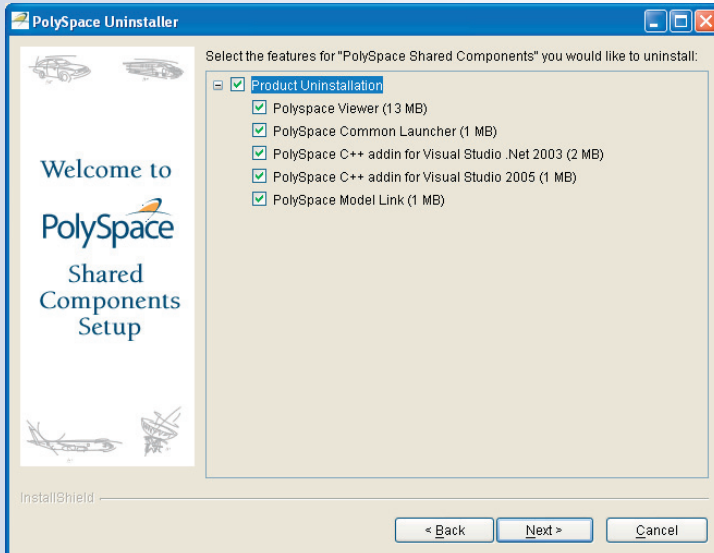


Figure 24:
Select components to uninstall

You can select components to remove during un-installation or call directly the "PolySpace Shared Components" un-installation procedure in "Add or Remove Programs". Here is the detail of the uninstall procedure for each component.

9.2.1. PolySpace Viewer uninstall

This will automatically uninstall the graphical interface used to review PolySpace analysis results.

9.2.2. Common Launcher uninstall

This will uninstall both the PolySpace in One Click feature and the Remote Launcher.

- PolySpace in One Click uninstall

You can uninstall PolySpace in One Click by:

1. Using the exit command in the PolySpace in On Click menu.
2. Executing the "msconfig" Windows command and unchecking the "PolySpaceInOneClick" radio button in the "Startup" Tab.

- Remote Launcher uninstall

The graphical un-installer detects whether the associated service is running and may propose to stop it by checking the "Stop the service (daemon) now" radio button.

If the service is running the Remote Launcher cannot be uninstalled.

You can un-install the Queue Manager Service independently on a Windows platform using a Command line prompt. Please execute the following command:

```
<PolySpaceInstallCommon>/RemoteLauncher/bin/RemovePolySpaceService.bat
```

To un-install Queue Manager daemon on a UNIX platform using a shell, please execute the following command:

```
<PolySpaceInstallCommon>/RemoteLauncher/setup-remote-launcher -remove-service
```

9.2.3. PolySpace C++ addin for Visual uninstall

It can be uninstalled in batch using the following command:

```
<PolySpaceInstallDir>/VisualInterface/UVisualInstall.bat
```


9.2.4. PolySpace Model Link SL and TL uninstall

To uninstall PolySpace Model Link TL and SL, select the corresponding option during the un-install of the "PolySpace Shared component".

A manual uninstall of the component would consist in:

- Removing the PolySpace directory `<MATLAB>/polyspace` inside each MATLAB installation
- Updating the configuration of PolySpace in the `<MATLAB>/Toolbox/local/matlabrc.m` file (search for `%% POLYSACE CONFIGURATION` inside).

Manual uninstall of PolySpace Model Link SL and TL is not recommended.

9.2.5. PolySpace UML Link RH uninstall

To uninstall PolySpace Model Link RH, select the corresponding option during the un-install of the "PolySpace Shared component". A manual uninstall of the component would consist in removing all lines added during the installation by PolySpace Installer in the `rhapsody.ini` and `site.prp` files.

Manual uninstall of PolySpace Model Link RH is not recommended.

9.3. Un-installation on Linux/UNIX

To un-install PolySpace under Linux or UNIX systems, please execute the following commands:

```
<PolySpaceInstallDir>/_uninst/uninstall
```

and

```
<PolySpaceInstallCommon>/_uninst/uninstall
```

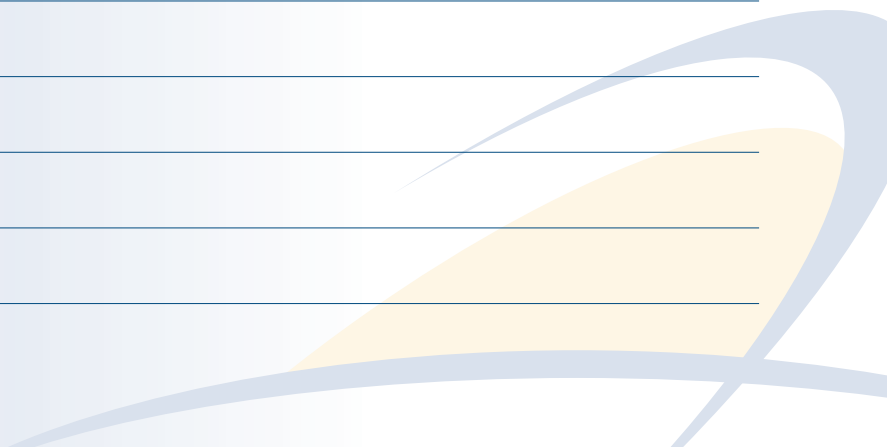
This will launch the graphical uninstall wizards.



A series of ten horizontal, dark blue lines spaced evenly down the page, serving as a template for writing or drawing.



A series of horizontal blue lines for writing, starting from the top of the page and extending across most of its width. A vertical blue line is positioned on the left side, creating a margin. The lines are evenly spaced and cover the majority of the page's content area.



PolySpace
TECHNOLOGIES

