Arm® Socrates™

Version 1.5

User Guide

101399_0105_00_en

Arm® Socrates™

User Guide

Release Information

Document History

Issue	Date	Confidentiality	Change
0100-00	25 July 2018	Confidential	First release
0102-00	23 November 2018	Confidential	Second release
0103-00	08 February 2019	Confidential	Third release
010301-00	18 March 2019	Confidential	Fourth release
010302-00	24 May 2019	Confidential	Fifth release
010303-00	16 July 2019	Confidential	Sixth release
010304-00	01 October 2019	Confidential	Seventh release
0104-00	06 February 2020	Confidential	Eighth release
0105-00	25 September 2020	Non-Confidential	Ninth release

Non-Confidential Proprietary Notice

This document is protected by copyright and other related rights and the practice or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm. **No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.**

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether implementations infringe any third party patents.

THIS DOCUMENT IS PROVIDED “AS IS”. ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, third party patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

If any of the provisions contained in these terms conflict with any of the provisions of any click through or signed written agreement covering this document with Arm, then the click through or signed written agreement prevails over and supersedes the conflicting provisions of these terms. This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is Final, that is for a developed product.

Web Address

developer.arm.com

Contents
Arm® Socrates™ User Guide

Preface

This preface introduces the *Arm® Socrates™ User Guide*.

It contains the following:

About this book

This User Guide is for the Arm Socrates™ IP Tooling platform.

Product revision status

The r*x*p*y* identifier indicates the revision status of the product described in this book, for example, r1p2, where:

r*x* Identifies the major revision of the product, for example, r1.

p*y* Identifies the minor revision or modification status of the product, for example, p2.

Intended audience

This guide is written for system designers, integrators, engineers, and programmers who are configuring or building Arm IP using the Arm Socrates™ platform.

Using this book

This book is organized into the following chapters:

Chapter 1 Introduction
Socrates enables you to browse, configure, (with configuration space validation), and build Arm IP, including Arm Interconnect IP. The generated RTL and IP-XACT can then be integrated into Arm-based systems. Read this chapter if you are new to Socrates and want a quick overview of the product.

Chapter 2 Setup and Maintenance Tasks
To ensure that you have the latest version of the product and IP Catalog, you must perform several setup tasks when you first run Socrates. Repeat these tasks periodically to stay up-to-date with the latest releases and functionality. Read this chapter if you need to keep Socrates up-to-date.

Chapter 3 Selecting, Configuring, and Building IP
Socrates provides several types of technical information for each supported Arm IP, and a standardized way to configure and build IP. You organize your workspace into projects, ensure that the relevant IP has been associated with a downloaded IP bundle, configure, and build. Read this chapter if you are new to Socrates and want to understand how the product works.

Chapter 4 Scripting API
The script-based API enables tasks to be executed in script form, and output generated. You can use the **Project Explorer** or the *Command Line Interface* (CLI) to run scripts that you have created.

Appendix A Command Line Interface
This appendix describes the *Command Line Interface* that Socrates provides to run workflows.

Appendix B Revisions
This appendix describes the changes between released issues of this book.

Glossary

The Arm® Glossary is a list of terms used in Arm documentation, together with definitions for those terms. The Arm Glossary does not contain terms that are industry standard unless the Arm meaning differs from the generally accepted meaning.

See the *Arm® Glossary* for more information.

Typographic conventions

italic
Introduces special terminology, denotes cross-references, and citations.

bold
Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.

`monospace`
Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.

`monospace`
Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.

`monospace italic`
Denotes arguments to monospace text where the argument is to be replaced by a specific value.

`monospace bold`
Denotes language keywords when used outside example code.

`<and>`
Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example:

```
MRC p15, 0, <Rd>, <CRn>, <CRm>, <Opcode_2>
```

SMALL CAPITALS
Used in body text for a few terms that have specific technical meanings, that are defined in the *Arm® Glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Additional reading

This book contains information that is specific to this product. See the following documents for other relevant information.

Arm publications

- *Arm® Socrates™ Installation Guide* (101400)

Feedback

Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

Feedback on content

If you have comments on content then send an e-mail to *errata@arm.com*. Give:

- The title *Arm Socrates User Guide*.
- The number 101399_0105_00_en.
- If applicable, the page number(s) to which your comments refer.
- A concise explanation of your comments.

Arm also welcomes general suggestions for additions and improvements.

Note

Arm tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

Chapter 1
Introduction

Socrates enables you to browse, configure, (with configuration space validation), and build Arm IP, including Arm Interconnect IP. The generated RTL and IP-XACT can then be integrated into Arm-based systems. Read this chapter if you are new to Socrates and want a quick overview of the product.

It contains the following sections:

1.1 About Socrates

Socrates IP Tooling platform is an environment for exploring, configuring, and building Arm IP ready for integration into a *System on Chip* (SoC).

The following examples show you some of the ways you can use Socrates.

Designing a SoC

If you want to:

- Investigate Arm IP features, properties, and configuration options
- Use this information to decide which IP to use in your system or subsystem design
- See which IP you already have licenses for
- Provide your procurement team with a detailed list of part numbers for the IP that you want to license

You can:

- Browse the **IP Catalog** and view information in **Socrates Help**
- Filter the **IP Catalog** by IP name or supported protocols
- Export information from the **Bill of Materials** about required and alternative part numbers

Managing IP bundles

If you want to:

- Ensure that the IP Catalog stays up-to-date with the latest release information
- Make finding the location of downloaded IP easier
- Update the **IP Catalog** entries so they are correctly associated with the latest IP bundles

You can:

- View notifications for product and **IP Catalog** updates
- Enter shared download locations for all licensed IP bundles
- Enable automatic refreshing of associations

Develop a system with Arm IP

If you want to:

- Understand IP configuration options, interfaces, and protocols
- Integrate IP into a system
- Build IP-XACT and Verilog output
- Automate the configuration and build procedures

You can:

- Browse the **IP Catalog** and view technical documentation in **Socrates Help**
- Organize and configure IP in the **Project Explorer**
- See validation and configuration errors in the **Create Configured IP** window
- Automatically build simple IP as part of the configuration procedure
- Use the *Command Line Interface* (CLI), and script-based API, for script-based solutions

Creating a high-level view of a SoC design

If you want to:

- Create a high-level view of your SoC design, containing configured IP instances and their high-level connections
- Choose the most suitable interconnect IP for your design, based on the high-level connectivity requirements of the other IP in your design
- Use this high-level view for further refinement and development of your design

You can:

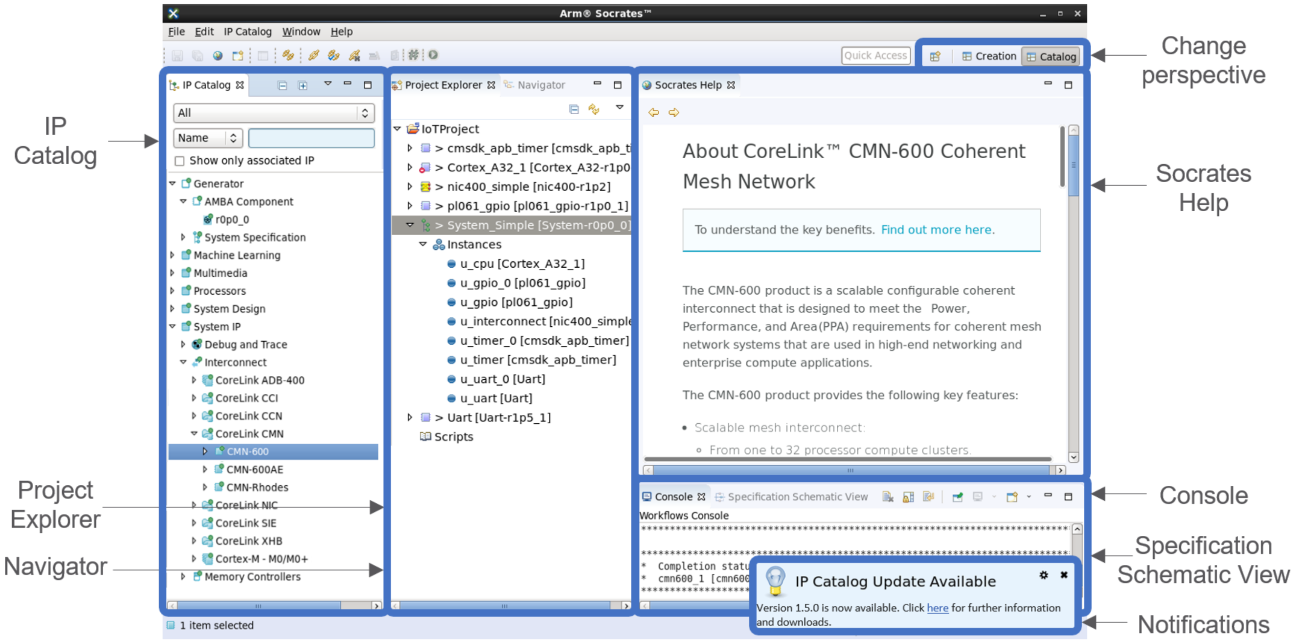
- Use the **System Specification**, from the **Generators** group in the **IP Catalog**, to create an empty System Specification
- Add instances of your required non-interconnect IP to the System Specification
- Use **Interconnect Assistant** to select, configure, and add the most suitable interconnect IP to the System Specification, automatically creating all the required high-level connections

1.2 Socrates at a glance

Socrates has a number of panels, known as views, arranged into two perspectives.

You can rearrange the panels to suit the way you work. To reset the default layouts, select **Window** > **Perspective** > **Reset Perspective...**.

The following figures show the default layout of views in Socrates.



About CoreLink™ CMN-600 Coherent Mesh Network

To understand the key benefits. Find out more here.

The CMN-600 product is a scalable configurable coherent interconnect that is designed to meet the Power, Performance, and Area(PPA) requirements for coherent mesh network systems that are used in high-end networking and enterprise compute applications.

The CMN-600 product provides the following key features:

- Scalable mesh interconnect:
  - From one to 32 processor compute clusters.

Workflows Console

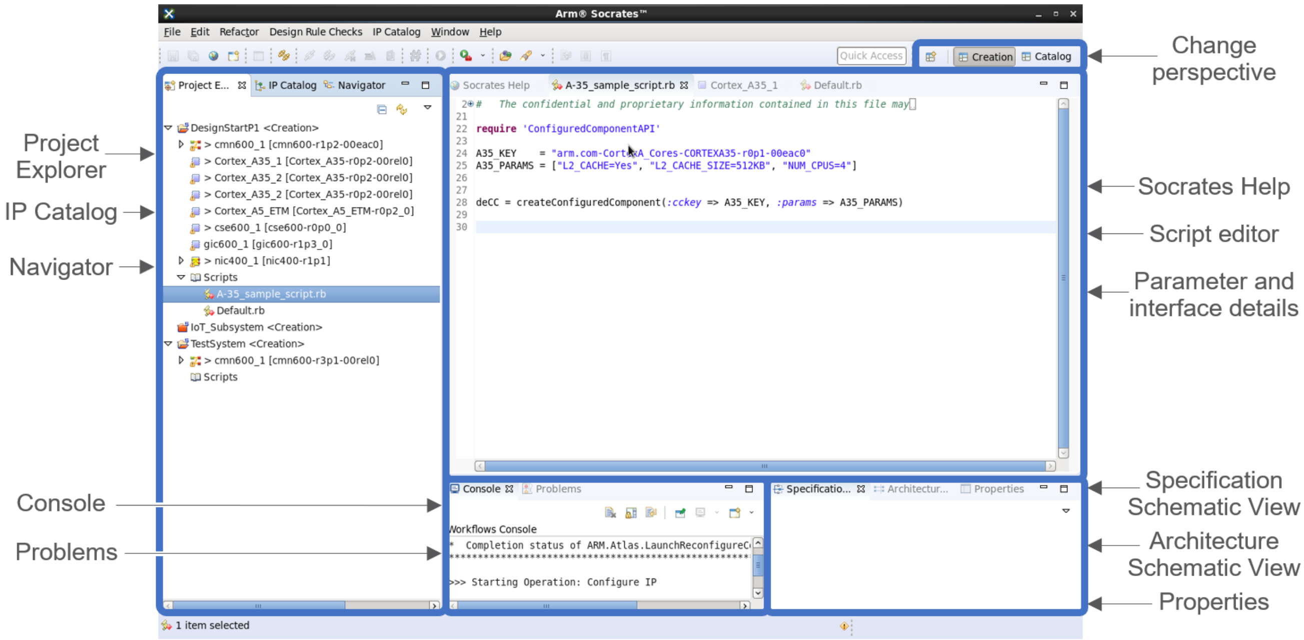
```
*****************************************************************************

*****************************************************************************
*  Completion statu
*  cmn600_1 [cmn600
*****************
```

IP Catalog Update Available

Version 1.5.0 is now available. Click here for further information and downloads.

Figure 1-1 Catalog perspective layout



```
#   The confidential and proprietary information contained in this file may

require 'ConfiguredComponentAPI'

A35_KEY    = "arm.com-CortexA_Cores-CORTEXA35-r0p1-00eac0"
A35_PARAMS = ["L2_CACHE=Yes", "L2_CACHE_SIZE=512KB", "NUM_CPUS=4"]


deCC = createConfiguredComponent(:cckey => A35_KEY, :params => A35_PARAMS)

```

Workflows Console

```
*  Completion status of ARM.Atlas.LaunchReconfigureC
*****************************************************

>>> Starting Operation: Configure IP
```

Figure 1-2 Creation perspective layout

The following table describes the basic functionality of the views.

Table 1-1 Description of Socrates views

Name	Description
Change perspective	Switch between the Catalog and Creation layouts.
IP Catalog	A hierarchical list of all the supported Arm IP products and versions. You can filter the list by group, name, protocol, or valid associations. See *3.1 Finding IP* on page 3-21.
Project Explorer	Projects and configured IP. See *3.3.1 Project Explorer* on page 3-23.
Navigator	File system view of projects.
Socrates Help	Detailed information about the selected IP.
Script editor	Use to edit scripts, see *4.1 API scripts* on page 4-32.
Parameter and interface details	Double-click an IP entry in the **Project Explorer** to see information about configuration and interfaces.
Console	Shows progress and status for a number of functions, for example, Checking Associations.
Problems	Details any problems found.
Specification Schematic View	Shows a graphical representation of the specification for an IP entry selected in the **Project Explorer**.
Architecture Schematic View	Shows a schematic of the µArchitecture for an IP entry selected in the **Project Explorer**.
Properties	Shows properties of the selected IP.
Notifications	Shows when updates are available for the product or the **IP Catalog**, see *2.2 Product updates* on page 2-18.

———— **Note** ————

Socrates is built on Eclipse, and not all of the Eclipse functionality is used or supported by Socrates. Therefore, if the functionality is not described in the Socrates documentation it might have unexpected behavior. For example, Socrates does not support the **Add project to working sets** option in the **File** > **Import** > **General** > **Existing Projects into Workspace** window.

————————————

1.3 Data collection in Socrates

Arm periodically collects anonymous information about the usage of our products in order to understand and analyze what components or features you are using with the goal to improve our products and your experience with them. Product usage analytics contain information such as system information, settings and usage of specific features of the product. You can enable or disable the feature in the product settings. Product usage analytics do not include any personal information.

Host information includes:

- Operating system name, version, and locale.
- Number of CPUs.
- Amount of physical memory.
- Screen resolution.
- Processor and GPU type.

Feature tracking information includes:

Table 1-2 Socrates analytics data points

Name	Description	Since
License type usage	• Tracked: — When Socrates is started (either directly, from the Socrates CLI, or from scripting.) — The type of license being used • Reported: Percentage of users using the different license types • Data type: Text • Send policy: Every invocation • Trigger points: On starting Socrates	Version 1.5

Disabling analytics collection

Analytics collection is enabled by default. You can disable it by using any of these methods:

- Select **Window** > **Preferences**, expand **Socrates**, then click **Analytics**. Clear the **Allow collection of product usage analytics** checkbox. Clearing this checkbox disables further analytics collection for this and subsequent Socrates sessions.
- Set the `ARM_DISABLE_ANALYTICS` environment variable to any value (including zero or an empty string). Setting this variable disables analytics collection for all Arm products running in that environment.
- Use the `--disable-analytics` option when you start Socrates from the Command Line Interface. This option disables analytics collection for the Socrates session. (Using `--help` to query the available command-line options does not trigger reporting.)

Chapter 2
Setup and Maintenance Tasks

To ensure that you have the latest version of the product and IP Catalog, you must perform several setup tasks when you first run Socrates. Repeat these tasks periodically to stay up-to-date with the latest releases and functionality. Read this chapter if you need to keep Socrates up-to-date.

Note

For details about installing Socrates, see *Arm® Socrates™ Installation Guide*.

It contains the following sections:

2.1 Workspaces

Socrates uses a workspace to store preferences and project references. When Socrates starts, you are asked to confirm the location that you want to use as a workspace. You can have multiple workspaces.

Procedure

To select a workspace:

1. Open the **Select a directory as workspace** window, either:
   - When Socrates starts, the window opens automatically.
   - When Socrates is running, select **File** > **Switch Workspace**.
2. In the **Select a directory as workspace** window, either:
   - **Browse…** to the folder you want to use.
   - Expand **Recent workspaces**, and select a previously used folder.
3. Optionally, to use the same workspace every time Socrates starts, select **Use this as default and do not ask again**.

2.2 Product updates

To ensure that you have the latest version of the product and the **IP Catalog**, you can install any available updates.

Every time Socrates starts, it automatically checks for the latest versions, and notifies you when they are available. You can view the available updates, start installing, and change the automatic setting in the **Updates** window.

To open the **Updates** window, either:

- Select **IP Catalog** > **Update IP Catalog**.
- In the **IP Catalog**, click **Preferences** ⚙ and select **Updates**.
- Select **Window** > **Preferences**, expand **IP Catalog**, and select **Updates**.
- Click the download link in the notification message.

The following table describes the functionality of the **Updates** window.

Table 2-1 Updates window functionality

Name	Description
Automatically check for product updates	Change the setting to disable automatic checking for updates and notification. Default is selected.
Check for updates	Use to manually check for updates when the automatic checking is turned off.
Updates section	Shows any available updates to the **IP Catalog** for current and newer versions of Socrates.
Install IP Catalog Updates	Launches the updated **IP Catalog** installation, follow the on-screen instruction.

Note

Socrates uses an environment variable (`ARM_SYSOC_UPDATE_NOTIFICATIONS`) to control whether the product update functionality is enabled. When this variable is set to `DISABLE`, all notifications and updates are disabled.

2.3 Licensed IP locations

Depending on your **Socrates** license, you can configure most IP without having a license for it. However, to build IP-XACT and Verilog output you must download a licensed IP bundle to your system and associate it with the corresponding entry in the **IP catalog**.

To download your licensed IP bundles:

1. Go to *https://connect.arm.com*, and log on using an account with the appropriate permissions.
2. Install each bundle according to the instructions in the specific IP documentation.

To keep associations up-to-date, you can set the download locations for your Arm IP and turn on automatic association in the **IP Library Path** window.

To open the **IP Library Path** window, either:

- Select **IP Catalog** > **IP Library Path Settings**.
- In the **IP Catalog**, click **Preferences** ⚙.
- Select **Window** > **Preferences**, expand **IP Catalog**, and select **IP Library Path**.

The following table describes the controls in the **IP Library Path** window that enable you to change the automatic association settings and set the IP bundle download locations.

Table 2-2 IP Library Path window functionality

Name	Description
Refresh associations on startup	Changes the setting to automatically associate **IP Catalog** entries with IP bundles that have been downloaded into the locations in `ARM_IP_LIBRARY_PATH`. Selected by default.
Overwrite existing associations on startup	When Socrates starts, this checkbox changes the setting to overwrite existing associations with the latest IP bundles in the download locations.
ARM_IP_LIBRARY_PATH	A colon (:) separated list of the download locations that you use for licensed Arm IP.
Overwrite existing associations	When an association is created manually, this checkbox changes the setting to overwrite existing associations with more recent IP bundles in the download locations.
Refresh	Check all associations for errors, create new associations, and overwrite existing associations with more recent IP bundles in the download locations.

Related tasks

3.2 Associating licensed Arm IP on page 3-22

Chapter 3
Selecting, Configuring, and Building IP

Socrates provides several types of technical information for each supported Arm IP, and a standardized way to configure and build IP. You organize your workspace into projects, ensure that the relevant IP has been associated with a downloaded IP bundle, configure, and build. Read this chapter if you are new to Socrates and want to understand how the product works.

It contains the following sections:

3.1 Finding IP

Use the **IP Catalog** to find the IP you need, and to access relevant information about that IP. You can filter the **IP Catalog** by group, name, protocol, or valid associations.

If the **IP Catalog** is not visible:

- Select **Window** > **Show View**, and select **IP Catalog**.

In the **IP Catalog**, you might see notification icons next to IP instances.

The following table describes the notification icons in the **IP Catalog**.

Table 3-1 IP Catalog notification icons

Name	Description
⚠	The IP instance has not been associated with a licensed IP bundle.
🟢⚠	The IP instance can be built but some configurations require additional IP feature bundles to build.
❌	There is an error with an existing association.
🟢	You can configure and build this IP instance with the license you are using.
🟥	You cannot configure or build this IP instance with the license you are using.

To show configuration guidance about each IP entry in the **IP Catalog**, select an IP entry. You can access further functionality using the **IP Catalog** menu.

To open the **IP Catalog** menu:

- Right-click an entry in the **IP Catalog**.

The following table describes the options on the **IP Catalog** menu.

Table 3-2 IP Catalog menu

Name	Description
Configure	Opens the **Create Configured IP** window, enabling you to reconfigure the selected IP.
Associate IP Bundle	Enables you to manually select a location of one or more licensed IP bundles, and associate with the selected IP entries in the **IP Catalog**.
Check Associations	Checks the current association for errors. Any errors that are found are highlighted.
Remove Associations	Removes the association of an IP entry with a licensed IP bundle.
Bill of Materials	Shows the **Bill of Materials** for the selected IP, that lists the part numbers for required IP, alternative versions, features, and dependencies.
Documentation	Shows the product documentation, if the entry has a valid association and documentation is included in the IP bundle.
Properties	Opens information about the selected entry, for example links to further information on *www.arm.com* and *developer.arm.com*.

3.2 Associating licensed Arm IP

Depending on your **Socrates** license, you can configure most IP without having a license for it. However, to build IP-XACT and Verilog output, you must associate the **IP Catalog** entry with a licensed IP bundle.

By default, Socrates automatically associates **IP Catalog** entries with IP bundles that have been downloaded to your system, as long as the download locations are listed in `ARM_IP_LIBRARY_PATH`. See *2.3 Licensed IP locations* on page 2-19.

Procedure

To manually associate an **IP Catalog** entry with a licensed IP bundle:

1. In the **IP Catalog**, select the folder or IP entry you want to associate with an IP bundle. If you select a folder, all entries within the folder are associated.
2. Right-click and select **Associate IP Bundle**.
3. Enter, or **Browse** to, the location of your downloaded bundle or bundles. This location can be the same as, or different to, the locations in `ARM_IP_LIBRARY_PATH`.
4. Optionally, select **Overwrite existing bundles**.
5. Click **Finish**.

3.3 Managing projects

A project is a container for all the design data that is required to produce a SoC product.

Configured IP instances are saved to a project as part of the configuration procedure, so you must have a least one project created before you can configure your IP. See *3.4 Configuring IP* on page 3-26.

This section contains the following subsections:
- *3.3.1 Project Explorer* on page 3-23.
- *3.3.2 Collaborative working* on page 3-24.

3.3.1 Project Explorer

Projects and configured IP are managed in the **Project Explorer** in the **Catalog** perspective.

If the **Project Explorer** is not visible:
- Select **Window** > **Show View**, and select **Project Explorer**.

In the **Project Explorer**, you can see notification icons next to any IP instances that have not been built, or associated with a downloaded IP bundle.

The following table describes the notification icons in the **Project Explorer**.

Table 3-3 Project Explorer notification icons

Name	Description
> (greater than symbol)	The IP instance has not been built.
⚠	The IP instance has not been associated with a licensed IP bundle.

You can access more functionality using the **Project Explorer** menu.

To open the **Project Explorer** menu:
- Right-click the **Project Explorer**, a project, or a configured IP instance, in the **Project Explorer**.

———— **Note** ————

When your configuration is selected in the Project Explorer, the Schematic Viewer shows only the AMBA Designer interfaces. If you want to see all the interfaces, then you must disable the **Show AMBA interfaces only** option.

————————————

The following table describes the options for projects on the **Project Explorer** menu.

Table 3-4 Project Explorer menu for projects

Name	Description
New Project...	Opens the **New Project** window, follow the on-screen instructions to create a project.
Bill of Materials	Shows the **Bill of Materials** for the selected project, that lists the part numbers for required IP, alternative versions, features, and dependencies.
Delete	Opens the **Delete Project** window, where you can delete a project, and optionally, all the project data from the file system. Deleting data cannot be undone.
Refresh	Forces a refresh of the **Project Explorer** information.
Properties	Opens information about the selected project, for example path variables, file locations, references to other projects, and details of any builders for the project.

Table 3-4 Project Explorer menu for projects (continued)

Name	Description
Open Project	Opens a project, where the underlying project data is saved in the workspace, but has not been loaded into the **Project Explorer**. Open projects are indicated with the open folder icon.
Close Project	Closes a project, removes the underlying project data from the **Project Explorer**. Closed projects are indicated with the closed folder icon.

Note

You can import projects that have been created in different workspaces, or by different people. You can use this function to work collaboratively on projects, see *3.3.2 Collaborative working* on page 3-24.

The following table describes the options for configured IP instances on the **Project Explorer** menu.

Table 3-5 Project Explorer menu for IP instances

Name	Description
New Project...	Opens the **New Project** window, follow the on-screen instructions to create a project.
Interconnect Assistant	Opens the **Interconnect Assistant**, enabling you to select, configure, and add an interconnect IP to the System Description.
Open	Opens parameter and interface details for the selected IP.
Reconfigure	Opens the **Create Configured IP** window, enabling you to reconfigure the selected IP.
Build	Starts the rendering of IP-XACT and Verilog output for the selected IP. Usually, the build is performed as an automatic part of the configuration procedure. However, some IP require this separate action. Where this is the case, it is described in the specific *User Guide* for that IP.
Bill of Materials	Shows the **Bill of Materials** for the selected IP, that lists the part numbers for required IP, alternative versions, features, and dependencies.
Delete	Deletes the selected IP, from the project and the file system.
Refresh	Forces a refresh of the **Project Explorer** information.
Properties	Opens file information for the selected IP, for example, path, filetype, location of file in the workspace, and permissions.

3.3.2 Collaborative working

Socrates facilitates collaborative working by enabling you to reuse IP configurations.

Sharing projects

You can use **File** > **Import** to import a project, and the contained IP, into your workspace. However, if you do not select **Copy projects into workspace**, you create a link to the original project. Any changes to one of the linked projects are reflected in the other.

This sharing of a project enables more than one person to work on the project.

The following is an example of how this feature can work:

1. You create a project, outside your workspace, in a shared location.
2. A colleague imports the project, ensuring **Copy projects into workspace** is not selected. They then add or reconfigure the IP.
3. You refresh the project and see the changed configurations.
4. Either of you can build or rebuild the IP.

Version control can be used on the shared location, as long as the **logical** directory in the project is excluded.

Sharing configured IP

You can export and import instances of configured IP between projects.

To export a configured IP instance .soc file:

1. In **Project Explorer**, right-click a configured IP instance, and select **Export**.
2. Expand **Socrates**, select **Export to Socrates Share (Collaboration)**, and click **Next**.
3. Select the configurations you want to export.
4. Browse to a location to save the configurations (as .soc files).
5. Click **Finish**.

To import a configured IP instance:

1. In **Project Explorer**, right-click a project, and select **Import**.
2. Expand **Socrates**, select **IP configurations**, and click **Next**.
3. Browse to the location of a saved configuration. Confirm the project that you want to import the configurations into and click **Next**.
4. Select the configurations you want to import and click **Finish**.

Sharing a configuration script

To export a configured IP to a script file:

1. In **Project Explorer**, right-click a configured IP instance, and select **Export**.
2. Expand **Socrates**, select **Export to Configure Script File**, and click **Next**.
3. Select the configurations you want to export.
4. Browse to a location to save the configurations (as .rb files).
5. Click **Finish**.

Note

A separate configuration file (.yml or .xml file) is generated for each configuration, and stored with the script file. If you copy a script file to a new location, you must also copy its associated configuration files to the same location.

To import a configuration script file:

1. In **Project Explorer**, right-click on **Scripts**.
2. Select **New Script...**.
3. Enable **Link to file in file system** and browse for the file location. (Make sure that the location contains both the script file and its associated configuration files.)
4. Click **Finish**.

Sharing configurations with Arm

To export a configured IP to send to Arm Support:

1. In **Project Explorer**, right-click a configured IP instance, and select **Export**.
2. Expand **Socrates**, select **Export to Arm Support**, and click **Next**.
3. Select the configurations you want to export.
4. Browse to a location to save the configurations (as .soc files).
5. Click **Finish**.

3.4 Configuring IP

In the **Create Configured IP** window, you can configure most IP, add the configured IP to your project, and usually, build the IP.

——— **Note** ———

You can use the Interconnect Assistant to help you choose and configure interconnect IP. See *3.6 Adding interconnect IP with Interconnect Assistant* on page 3-28.

———————————

To show configuration guidance in **Socrates Help**:

- In the **IP Catalog**, select an IP entry.

To open the **Create Configured IP** window, either:

- Double-click an IP entry in the **IP Catalog**.
- Right-click an IP entry in the **IP Catalog**, and select **Configure**.

If you are prompted to open the **Creation** perspective, you must use a different configuration procedure, see the Socrates User Guide for the specific IP.

Most builds are an automatic part of the configuration procedure, however, if the IP entry has a synthesis step, the build must be done manually, see *3.5 Building IP* on page 3-27.

Prerequisites

- You have created at least one project, see *3.3 Managing projects* on page 3-23.
- You have associated the IP with a licensed IP bundle, see *3.2 Associating licensed Arm IP* on page 3-22.

Procedure

To configure an IP instance:

1. In the **Create Configured IP** window, edit the parameters for your chosen IP.
2. If the selected IP has more than one page of parameters, complete each page, and click **Next**.
3. In **Project**, select the destination project for the configured IP.
4. In **Configured IP name suffix**, enter a name suffix or use the default.
5. Optionally, to override the build process, clear **Build Automatically**.
6. Click **Finish**. An instance of the IP with the specified parameter configuration is added to the project in the **Project Explorer**.

To reconfigure an IP instance:

- Right-click an entry in the **Project Explorer**, and select **Reconfigure**.

3.5 Building IP

Building a configured IP renders IP-XACT and Verilog output into the project directory.

Some IP is built automatically after configuration. For complex IP, or if you want to run the build step manually from the editor, use the **Build** button.

3.6 Adding interconnect IP with Interconnect Assistant

Use the Interconnect Assistant to choose and add the most suitable interconnect IP to your SoC design.

This section contains the following subsections:

- *3.6.1 Component Generators* on page 3-28.
- *3.6.2 Create an empty System Specification* on page 3-28.
- *3.6.3 Add non-interconnect IP to the System Specification* on page 3-29.
- *3.6.4 Select and connect your IP with Interconnect Assistant* on page 3-29.

3.6.1 Component Generators

Component generators can help you to design your SoC and configure your Arm IP.

System Specification Generator

Use the System Specification generator to create an empty System Specification and generate a high-level view of a SoC design. This view shows the configured IP instances in the design and the high-level connections between them, providing a starting point for further development. See *3.6.2 Create an empty System Specification* on page 3-28.

A System Specification:

- Contains only instances and high-level connections
- Is used for configuration only
- Cannot be associated with Arm IP
- Cannot be built

You can use System Specifications in multiple ways. Depending on the complexity of your SoC design, you may want to use multiple System Specifications, for example:

- To explore different design alternatives
- If different groups of IP in the design require different types of interconnect IP

Component generators are listed under the **[Generators]** group in the **IP Catalog**. The following generators are available.

AMBA Component Generator

Use the AMBA Component generator to create a mock component that you can configure with AMBA interfaces and protocols. This component provides no other functionality.

Note

The component can, for example, represent an instance of third-party IP in a System Specification. See *3.6.3 Add non-interconnect IP to the System Specification* on page 3-29.

An AMBA component:

- Contains only AMBA interfaces (master and slave, AXI, ACE, AHB, CHI, APB)
- Is used for configuration only
- Cannot be associated with Arm IP
- Cannot be built

3.6.2 Create an empty System Specification

An empty System Specification is the starting point for your design.

Prerequisites

Ensure you have a new or existing project. See *3.3 Managing projects* on page 3-23.

Procedure

1. In the **Generators** group in the **IP Catalog**, double-click the **System Specification**.
2. In the **Create Configured IP** window:
    a. Select your new project.
    b. In **Configured IP name suffix**, enter a name suffix or use the default.
    c. Click **Finish**.

An empty System Specification is added to the project.

Next Steps

3.6.3 Add non-interconnect IP to the System Specification on page 3-29.

3.6.3 Add non-interconnect IP to the System Specification

Create the instances of each non-interconnect IP required by your design.

Prerequisites

Ensure that you have a project containing:

- A System Specification. See *3.6.2 Create an empty System Specification* on page 3-28.
- The non-interconnect IP that you want to use in your system. See *Chapter 3 Selecting, Configuring, and Building IP* on page 3-20.

  ——— **Note** ———

  Use an AMBA component to represent third-party IP that you need. See *3.6.1 Component Generators* on page 3-28.

  ———————————

Procedure

1. In **Project Explorer**, select the System Specification.
2. In the **Instances** tab, click ✚.
3. In the **Create Instances** window:
    a. Select each **IP Configuration** that you want to use.
    b. For each selected **IP Configuration**, enter the number of instances that you want to create.
    c. Click **Next**.
4. In the **Instances** window, review and change instance names, or use the default values.
5. Click **Finish**, then save the project.

The IP instances are listed in the System Specification. You can also see the IP instances and their interfaces in the **Architecture Schematic View**.

Next Steps

3.6.4 Select and connect your IP with Interconnect Assistant on page 3-29.

3.6.4 Select and connect your IP with Interconnect Assistant

The **Interconnect Assistant** simplifies the task of choosing the most suitable interconnect IP to add to your System Specification, automatically creating all the required high-level connections.

Firstly, the **Interconnect Assistant**:

- Determines the number and types of interfaces presented by all the IP instances in your system
- Ranks the available interconnect IPs in order of their support for the IP interfaces

Secondly, you choose:

- The interconnect IP to use
- The master and slave interfaces to connect

Finally, the **Interconnect Assistant**:

- Creates, configures, and instantiates the chosen interconnect IP.
- Creates the specified high-level connections between the interconnect IP instance and the other IP instances.

To open **Interconnect Assistant** for a System Specification, in **Project Explorer**, either:

- Select the System Specification, and click **Open Interconnect Assistant Wizard** on the toolbar.
- Right-click the System Specification, and choose **Interconnect Assistant**.

Prerequisites

Ensure you have a project containing the instances of the non-interconnect IP that you want to use in your system. See *3.6.3 Add non-interconnect IP to the System Specification* on page 3-29.

Procedure

1. Open **Interconnect Assistant**.
2. In **Select Interconnect**, select the interconnect IP that you want to use.

   The window shows the total number of interfaces used by all the IP instances in the system that each interconnect IP supports. The number is shown separately for interfaces that are currently connected or not connected in the system. The interconnect IP are listed in descending order of the total number of interfaces that each can support.
3. In **Configure slave interfaces**, select each master interface on an IP instance that you want to connect to a matching slave interface on the interconnect IP instance.

   The window shows:

   - All compatible interfaces present in the system (grouped by their parent IP instances)
   - The interface types that are available in this interconnect IP, and any constraints that apply to each interface type

     ——— **Note** ———

     If your selections violate a constraint, an appropriate warning or error marker is displayed.

     ———————————

4. In **Configure master interfaces**, repeat this process to select master interface connections for the interconnect IP.
5. In **Create interconnect**, specify the configuration name suffix and instance name for the interconnect IP.
6. Click **Finish**.

You can see the connected IP in the **Architecture Schematic View**.

——— **Note** ———

Based on its analysis of the configured IP in the System Specification, the **Interconnect Assistant** tries to provide sensible configuration values for each chosen interconnect IP. (If applicable, default memory maps, paths, and non-specified interfaces are created.) However, we cannot guarantee that any of these values are the most appropriate for your SoC design. You must review the generated configuration to ensure that it meets your specific requirements.

———————————

Chapter 4
Scripting API

The script-based API enables tasks to be executed in script form, and output generated. You can use the **Project Explorer** or the *Command Line Interface* (CLI) to run scripts that you have created.

It contains the following sections:

4.1 API scripts

Socrates IP Tooling platform supports scripts in Ruby, Tcl, and Python.

To create, edit, and run API scripts, you must be in the **Creation** perspective.

The following tasks can be accessed in **Project Explorer** by expanding the appropriate project and using the **Scripts** node:

- To create a script, right-click **Scripts** and select **New Script**.
- To open an existing script in the editor, double-click the script in the **Project Explorer**.
- To run a script, right-click the script in the **Project Explorer** and select **Run Script**.
- To run a script in Ruby, in the script editor, click **Run Script** ▶, press **Ctrl+r**.

4.2 Configured Component API

The Configured Component API creates, configures, and builds configurable components.

To enable the Configured Component API in a script use:

```
require ‘ConfiguredComponentAPI’
```

The API uses the `HLS` methods to create the IP Specification.

Note

To use IP bundles in the sample scripts, associate the appropriate IP bundles.

This section contains the following subsection:

- *4.2.1 Cortex-A35 sample configure and build script* on page 4-33.

4.2.1 Cortex-A35 sample configure and build script

This sample script creates an instance of a Cortex®-A35 core, with the specified parameters, using the *Configured Component* API.

To open the sample script:

1. Open the **Creation** perspective.
2. In **Project Explorer**, expand your project and right-click on the **Scripts** node.
3. Select **New Script...**.
4. In the dialog, select **Sample Cortex A35 Configure and Build**.

The **IP Catalog** *Configured Component* key is used to identify which component to create. Any unspecified parameters take on their default value.

1. Enable the *Configured Component* API:

   ```
   require 'ConfiguredComponentAPI'
   ```

2. Define the *Configured Component* key from the IP Catalog, and the build parameters for the *Configured Component*. You might need to modify the supplied code to match the following parameters:

   ```
   A35_KEY = "arm.com-CortexA_Cores-CORTEXA35-r0p1-00eac0"
   A35_PARAMS = ["L2_CACHE=Yes", "L2_CACHE_SIZE=512KB", "NUM_CPUS=4", "CRYPTO=No",
   "NEON_FP=No"]
   ```

3. Create the Configured Component with the specified key and parameters:

   ```
   deCC = createConfiguredComponent(:cckey => A35_KEY, :params => A35_PARAMS)
   ```

The *Configured Component* is automatically built and is located in the project folder in the **Project Explorer**. The other files that the build generates are located in the project folder in the workspace directory of the user.

The `createConfiguredComponent` method returns the *Configured Component* design element, which is stored as `deCC`. Query this for more information on the built Configured Component.

For more information on the *Configured Component* API methods, see the Socrates Ruby API help in **Help** > **Help Contents** > **Arm Socrates Ruby API Manual** > **Ruby Configured Component API**.

4.3 Extra utility API

An extra utility API is available that supports a set of methods to create a Socrates script.

To enable the Ruby Utility API use:

```
require 'RubyUtilityAPI'
```

For more information, see the Ruby API help by selecting **Help** > **Help Contents** > **Arm Socrates Ruby API Manual**.

Appendix A
Command Line Interface

This appendix describes the *Command Line Interface* that Socrates provides to run workflows.

It contains the following section:

A.1 Basic commands for Socrates CLI

You can use the CLI to run commands for much of the functionality in Socrates.

Use the `--help` option to get more comprehensive help on the CLI:

```
socrates_cli --help
```

——— **Note** ———

For batch processing, the same composite workflow of CLI commands can be called several times during a single call using variable parameter values. The parameters that are used must be in a `<name>=<value>` pair, and the composite workflow being executed must understand them.

For example, `socrates_cli --project <project name> --flow <workflow name> parameter1=a --flow <workflow name> parameter1=b`.

———————

Commands are run against the default workspace in `/home/<user>/armSocrates/workspace/`. To run commands against a different workspace, explicitly refer to it using the parameter `-data /<path to workspace>/`.

The following table describes some of the commands that can be used on the CLI:

Table A-1 CLI commands

Operation	Description	Command line usage
Select license option	Set the license option, either `socrates`, `system_builder`, or `socrates_flexibleaccess_ms`.	`socrates_cli --license <licensing feature>`
Disable product usage analytics	This operation disables analytics collection for this Socrates session (only).	`socrates_cli --disable-analytics`
List catalog entries	This operation lists the current content of the IP catalog. If `includeStatus` is true, then the association status of each entry is also printed. Use the optional argument `includeKeys` to return the `ConfigurableKey` used when creating a configured component.	`socrates_cli --flow list.catalog.entries [includeStatus=true|false] [includeKeys=true|false]`
Add new Catalog entries	This operation adds new entries to the IP Catalog. `newEntryArchive` refers to an archive file containing the new catalog entries. ——— **Note** ——— This action requires the installation to be writable in the installation environment. ———————	`socrates_cli --flow add.catalog.entries newEntryArchive=<path and file name to archive file from which new entries are extracted from>`
Update the IP catalog to the latest version	This operation installs all updates from the latest version of the IP Catalog, and updates the IP Catalog version number to the latest version.	`socrates_cli --flow update.ip.catalog`

Table A-1 CLI commands (continued)

Operation	Description	Command line usage

Associate an IP bundle	This operation attempts to find catalog entry associations in the specified directory. If `overwrite` is true, then existing associations are overwritten.
	The `includeNewEntries` searches for new IP to add to the IP Catalog.
		`socrates_cli --flow associate.bundle.ip`
		`bundleDirectory=<directory location of bundle to`
		`be associated>`
		`[overwrite=true|false]`
		`[includeNewEntries=true|false]`

Create a project	This operation creates a project in the workspace.	`socrates_cli --project <new project name>`
		`--flow AddNewProject`

Create a configured component	This operation attempts to create a new configured component that is based on the configurable specified. The new configured component has its parameters set to the parameters that the `parameters` argument specifies.	`socrates_cli --project <existing project name>`
		`--flow create.configured.component`
		`configurableKey=<VLNV of the configurable`
		`component>`
		`[configuredComponentNameSuffix=<a name suffix for`
		`the configured component>]`
		`[parameters="parameter=value,parameter=value"]`
		`[subSystem="subsystem name"]`
		`[disableAutoBuild=true|false]`

Build a configured component	This operation executes the `BuildOperationReference` on the specified configured component.	`socrates_cli --project <existing project name>`
		`--flow build.configured.component`
		`configuredComponentName=<a name for the configured`
		`component>`
		`[subSystem=<sub system name>]`

Run a script	This operation executes the specified script as a scripted workflow.	`socrates_cli --project <existing project name>`
		`--flow RunScript`
		`ScriptFile=<location to script>`

Appendix B
Revisions

This appendix describes the changes between released issues of this book.

It contains the following section:

- *B.1 Revisions* on page Appx-B-39.

B.1 Revisions

This appendix describes changes between released issues of this book.

Table B-1 Issue 0101-00

Change	Location	Affects
First release	-	-

Table B-2 Differences between issue 0101-00 and issue 0102-00

Change	Location	Affects
Clarification of instructions to open build script.	*4.2.1 Cortex-A35 sample configure and build script* on page 4-33	Version 1.2 and onwards.

Table B-3 Differences between issue 0102-00 and issue 0103-00

Change	Location	Affects
Added examples of the different ways to use Socrates.	*1.1 About Socrates* on page 1-11	Version 1.3 and onwards.
Added screenshots and descriptions of perspectives and views.	*1.2 Socrates at a glance* on page 1-13	Version 1.3 and onwards.
Added tasks that are needed for setup and maintenance.	*Chapter 2 Setup and Maintenance Tasks* on page 2-16	Version 1.3 and onwards.
Added details about automatic updates for product and **IP Catalog**.	*2.2 Product updates* on page 2-18	Version 1.3 and onwards.
Added a description of **IP Catalog** functionality, including new filtering, and association checking.	*3.1 Finding IP* on page 3-21	Version 1.3 and onwards.
Added a description of **Project Explorer** functionality, including notification indicators.	*3.3.1 Project Explorer* on page 3-23	Version 1.3 and onwards.
Added a description of **Bill of Materials** functionality.	*3.1 Finding IP* on page 3-21 and *3.3.1 Project Explorer* on page 3-23	Version 1.3 and onwards.
Added suggestions for collaborative working on projects.	*3.3.2 Collaborative working* on page 3-24	All versions.

Table B-4 Differences between issue 0103-00 and issue 010301-00

Change	Location	Affects
Updated to 1.3.1 release.	-	-

Table B-5 Differences between issue 010301-00 and issue 010302-00

Change	Location	Affects
Updated to 1.3.2 release.	-	-

Table B-6 Differences between issue 010302-00 and issue 010303-00

Change	Location	Affects
Updated to include *Group* filter on **IP Catalog**.	*1.2 Socrates at a glance* on page 1-13, and *3.1 Finding IP* on page 3-21.	Version 1.3.3 and onwards.

Table B-7 Differences between issue 010303-00 and issue 010304-00

Change	Location	Affects
Updated to 1.3.4 release.	-	-

Table B-8 Differences between issue 010304-00 and issue 0104-00

Change	Location	Affects
Added information about support for Eclipse functionality.	*1.2 Socrates at a glance* on page 1-13	Version 1.4 and onwards.
Added alternative ways to open the **Updates** window.	*2.2 Product updates* on page 2-18	Version 1.4 and onwards.
Added alternative ways to open the **IP Library Path** window.	*2.3 Licensed IP locations* on page 2-19	Version 1.4 and onwards.
Added an icon to the **IP Catalog Notification icons** table regarding products where bundles need to be merged.	*3.1 Finding IP* on page 3-21	Version 1.4 and onwards.
Added information about sharing a configuration script and sharing configurations with Arm.	*3.3.2 Collaborative working* on page 3-24	Version 1.4 and onwards.

Table B-9 Differences between issue 0104-00 and issue 0105-00

Change	Location	Affects
First Non-Confidential release	-	Version 1.5 and onwards.
Added an example of how to use Socrates to create a high-level view of a SoC design.	*1.1 About Socrates* on page 1-11	Version 1.5 and onwards.
Added information about product usage analytics - what is collected, and how to disable collection.	*1.3 Data collection in Socrates* on page 1-15 and *A.1 Basic commands for Socrates CLI* on page Appx-A-36	Version 1.5 and onwards.
Added how to start **Interconnect Assistant** from **Project Explorer**.	*Table 3-5 Project Explorer menu for IP instances* on page 3-24	Version 1.5 and onwards.
Added how to use **Interconnect Assistant** to choose and add the most suitable interconnect IP a SoC design.	*3.6 Adding interconnect IP with Interconnect Assistant* on page 3-28	Version 1.5 and onwards.
Added CLI command to update the IP Catalog to the latest version.	*A.1 Basic commands for Socrates CLI* on page Appx-A-36	Version 1.5 and onwards.