

Technical Document

Sedona Framework TXS Sedona Tools Guide

October 18, 2012

Powered by
sedona
FRAMEWORK®

Powered by
niagara^{AX}
FRAMEWORK®

Sedona Framework TXS Sedona Tools Guide

Confidentiality Notice

The information contained in this document is confidential information of Tridium, Inc., a Delaware corporation ("Tridium"). Such information, and the software described herein, is furnished under a license agreement and may be used only in accordance with that agreement.

The information contained in this document is provided solely for use by Tridium employees, licensees, and system owners; and, except as permitted under the below copyright notice, is not to be released to, or reproduced for, anyone else.

While every effort has been made to assure the accuracy of this document, Tridium is not responsible for damages of any kind, including without limitation consequential damages, arising from the application of the information contained herein. Information and specifications published here are current as of the date of this publication and are subject to change without notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia.

Trademark Notice

BACnet and ASHRAE are registered trademarks of American Society of Heating, Refrigerating and Air-Conditioning Engineers. Microsoft, Excel, Internet Explorer, Windows, Windows Vista, Windows Server, and Visio are registered trademarks of Microsoft Corporation. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Mozilla and Firefox are registered trademarks of the Mozilla Foundation. Echelon, LON, LonMark, LonTalk, and LonWorks are registered trademarks of Echelon Corporation. Tridium, JACE, Niagara Framework, Niagara^{AX} Framework, and Sedona Framework are registered trademarks, and Workbench, WorkPlace^{AX}, and ^{AX}Supervisor, are trademarks of Tridium Inc. All other product names and services mentioned in this publication that is known to be trademarks, registered trademarks, or service marks are the property of their respective owners.

Copyright and Patent Notice

This document may be copied by parties who are authorized to distribute Tridium products in connection with distribution of those products, subject to the contracts that authorize such distribution. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Tridium, Inc.

Copyright © 2012 Tridium, Inc.

All rights reserved. The product(s) described herein may be covered by one or more U.S or foreign patents of Tridium.

CONTENTS

Preface	iii
About this document	iii
Sedona Tools terms	iii
Related documentation	v
Document change log	v
 About The Sedona Tools	 1-1
Changes to provisioning tools for TXS 1.2	1-1
About the Sedona Environment Manager	1-2
About the Sedona Tools and Schema	1-2
Accessing the Sedona Tools	1-2
Sedona Tools overview	1-3
Schema overview	1-4
 About The Kit Manager	 2-1
About the Kit Manager view	2-1
 About The Application Manager	 3-1
About the Get and Put operations	3-1
 About The Backup/Restore Tool	 4-1
About the Backup and Restore operations	4-1
 Common Tasks	 5-1
<i>Make kit selection changes</i>	<i>5-1</i>
<i>Get the app from a device</i>	<i>5-2</i>
<i>Put an application on a device</i>	<i>5-4</i>
<i>Create a backup</i>	<i>5-6</i>
<i>Restore a backup</i>	<i>5-7</i>
 FAQs and Troubleshooting	 A-1
Frequently asked questions	A-1
Troubleshooting tips	A-1
Adding a kit operation fails	A-2
Kit manager fails to load	A-2
Provisioning fails during backup	A-2
Provisioning fails when compiling code on JACE-2 series platform	A-2
Provisioning fails when Sedona Environment not configured on platform	A-3

Components and Plugins B-1

Sedona Tools components B-1

Sedona Tools plugins (views) B-1

 Sedona Tools viewB-1

 Backup/Restore Tool viewB-1

 Application Manager viewB-2

 Kit Manager viewB-2

PREFACE

Preface

This documents the Sedona Framework TXS Sedona Tools available after making a connection to a Sedona Device, when using either NiagaraAX Workbench 3.7 or later with Sedona Framework TXS 1.2, or when using Sedona Framework Workbench 1.2. Once connected to the device via Sox, both Workbench products provide the identical Sedona Tools for provisioning the Sedona Framework device.

It is important to note the different methods by which you can provision a Sedona Framework device. Sedona Framework TXS 1.2 allows you to provision a device through the *JACE* via the Sedona Tools, found under the device in the Sedona Network or Sedona Jen6lp Network. In this case, all of the necessary kits, manifests, and PAR files must be stored *on the station*. Provisioning is also possible through a direct *sox session-based* connection to a device, as is done in the TXS 1.1 release, and is still supported in TXS 1.2.

Note: *In order to provision a Sedona Framework enabled device using a JACE-2 series platform, the JACE needs to be licensed for the expanded 128MB of memory (“maxHeap” license feature). If you have a JACE that does not have this feature, contact your product vendor to determine how to add this requirement.*

This preface has the following sections:

- [About this document](#)
- [Sedona Tools terms](#)
- [Related documentation](#)
- [Document change log](#)

About this document

This document describes the Sedona Framework TXS 1.2 provisioning tools available when connected to a Sedona Framework device and has the following main sections:

- [“About The Sedona Tools”](#)
Explains changes to the provisioning tools for TXS 1.2, the Sedona Tools view and describes the Sedona Tools view and schema.
- [“About The Application Manager”](#)
Explains the App Manager tool and describes the procedures for copying an app from or to a Sedona Framework device.
- [“About The Backup/Restore Tool”](#)
Explains the Backup/Restore tool and describes the procedures for backing-up and restoring Sedona Framework device components.
- [“About The Kit Manager”](#)
Explains the Kit Manager tool and describes the procedure for making kit selection changes on a Sedona Framework device.
- [“FAQs and Troubleshooting”](#)
Provides frequently asked questions and troubleshooting tips.

Sedona Tools terms

The following is a list of terms and abbreviations used in this document when describing Sedona Framework TXS Sedona Tools and Sedona Framework devices. Complete Sedona Framework documentation is available at <http://sedonadev.org>. For general NiagaraAX terms, see the Glossary in the *Niagara AX User Guide*.

app For application. The app in a Sedona Framework device is its collection of Sedona Framework components, including links between them, plus configuration properties.

bundle Sedona Framework TXS software is distributed in a “bundle”. A bundle is a special image (set of files) that is available from Niagara-Central (www.niagara-central.com) or from Sedona OEM manufacturers. You use the Sedona Installer tool in Workbench to install a bundle. For complete details, refer to the *NiagaraAX Sedona Installer Guide*.

DASP For Datagram Authenticated Session Protocol. This is the low-level, session-based protocol that Sox utilizes.

kit Sedona Framework kits are the basic unit of modularity of Sedona Framework software, encapsulating code, types, and metadata. A kit is analogous to a module on a NiagaraAX platform. The app in a device instantiates components and services contained in its installed kits. You must have the appropriate kits available on your Workbench to change the kits on a device. Sedona Framework Sedona Tools in Workbench include a “Kit Manager” view to manage kits on a Sedona Framework device.

manifest Each kit has a corresponding manifest, with all metadata needed by tools like Workbench to connect via Sox to a Sedona Framework device, and for a station to support Sedona Framework proxy points in that device. Manifest files are compact XML files, named using a kitName-checksum convention similar to kit files, but with an .xml extension. To manage kit manifests on a Niagara AX host (JACE or Supervisor), right click on a Sedona Network node in the nav tree and select the Manifest Manager view. You can manage manifests on your Workbench host by using the Sedona Manifest Manager view in Workbench tools. For more details see the *Sedona Manifest Manager* document.

PAR For Platform Archive, or PAR file. This is a zip file with a .par extension that provides a way of organizing various files and metadata about a platform into a single entity. The term “platform” is used to refer to the Sedona capabilities of a given device or class of devices. It does not specify what kits are loaded onto the device at a given time, however it does define which kits are supported (i.e. that may be added to the device if desired). The platform archive contains an XML file that describes the platform. A PAR file may also contain a device simulator SVM. Par files can be imported using the Sedona Installer tool in Workbench.

SAB For Sedona App Binary. This is a compact binary representation of a Sedona Framework app suitable for storage and execution on a Sedona Framework device. One of two file formats (.sab) for a Sedona Framework app.

SAX For Sedona App XML. This is a simple XML representation of a Sedona Framework app that is easily generated and consumed by Sedona Framework software tools. One of two file formats (.sax) for a Sedona Framework app.

schema The specific collection of Sedona Framework kits installed in a device (each at some particular checksum revision) is called a schema. It also describes the set of kits required by a specific Sedona app. The schema of the app must be compatible with the schema of the device where it is to run. You can see the schema table when you access the Sedona Tools view when Sox-connected to a device.

scode An scode image is a binary file containing the kits of a particular schema compiled together. It is executed by the Sedona Virtual Machine (SVM). The Kit Manager tool is used to modify the kits.scode on a Sedona Framework device.

SEDONA HOME The term SEDONA HOME is a system “variable” for the installation-specific directory of your Workbench computer’s “root” folder for Sedona Framework files used in Sox connections from Workbench. To see this location, search for “sedona.home” in the `!lib/system.properties` file. When using the Sedona Installer to install a bundle, the SEDONA HOME system variable is set correctly for you.

Sox Sox is the standard protocol used to communicate with Sedona Framework devices. It runs over UDP via the lower-level DASP protocol. Workbench always uses Sox to open a connection to a Sedona Framework device. A Niagara station also uses Sox to discover as well as read and write to Sedona Framework proxy points.

SVM The SVM (Sedona Virtual Machine or VM) executes the compiled scode image in a device along with its app.

Related documentation

NiagaraAX Sedona Installer Guide

NiagaraAX Sedona Networks Guide

Document change log

Updates (changes/additions) to this *Niagara^{AX} Sedona Framework TXS Sedona Tools Guide* document are listed below.

- Publication: October 18, 2012
Initial document.

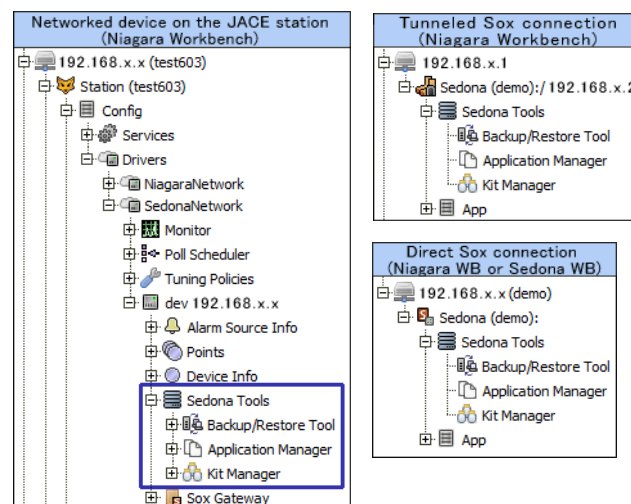
CHAPTER 1

About The Sedona Tools

The Sedona Tools are used to provision a single Sedona Device. In TXS 1.2, you can access a networked Sedona Device via a Fox connection to a JACE station and use the provisioning tools there. An alternative is to access a Sedona Device via a Sox-session connection (a direct Sox connection using either Sedona Workbench or Niagara Workbench or a tunneled-Sox connection using Niagara Workbench only).

In either case, the provisioning tools are the same; however, the Sedona environment that you are using is not. When accessing a networked device through a station on the JACE, all of the environment files *reside on that JACE*, as shown on the left side of [Figure 1-1](#). This differs when using a Sox session connection (direct Sox or Sox-tunneled) where the environment files *reside on the Workbench host*, as shown on the right side of [Figure 1-1](#).

Figure 1-1 Nav Tree shows Sedona Tools located under networked device on the JACE station (left) and under the device on the Workbench host in Sox session connections (right)



The following topics are covered in this section:

- [Changes to provisioning tools for TXS 1.2](#)
- [About the Sedona Tools and Schema](#)

Changes to provisioning tools for TXS 1.2

If you are familiar with Sedona TXS 1.1, you may find this summary of changes useful. Most notably, the name for the tools has changed from Sox Tools to “Sedona Tools”. One of the individual tools had a name change as well, App Manager is now named “Application Manager”.

One of the major enhancements in Sedona Framework TXS 1.2, is the added ability of the Niagara station backup to include Sedona device files such as App, Kit and PAR files. This ensures the presence of all of the files that your application requires to run if deployed on another device.

Another enhancement in TXS 1.2 allows you to provision a device through the JACE via a Sedona Network or Sedona Jen6lp Network. The provisioning tools are available under each networked device, as shown in [Figure 1-1](#) (left side).

Note: Provisioning a device through the JACE requires you to first configure the JACE controller via a platform connection, using the "Sedona Environment Manager" view.

A few changes to the provisioning tools have been introduced for the TXS 1.2 release to better allow the tools to operate without knowledge of whether you are provisioning from a Sedona environment in a Sox session or from a networked station. The tools that perform the get/put operations, (Application Manager) and the backup/restore operations (Backup/Restore Tool) have been modified to standardize the store and source locations shown for the files during these operations.

To provision a Sedona device when connected to a station on the JACE, using the Sedona Tools under a *networked device*, the saved apps (saved via the get application operation) and backups are always saved in the file space for that station. The location of the saved files follows a fixed convention, starting with a ^sedona/store/SedonaNetwork_Name/SedonaDevice_Name location followed by either the /apps or /backups folder depending on which operation you are performing. This convention ensures that the apps and backups are always available and included in a station copy.

When Sox tunneling through the JACE to a device using Niagara Workbench, or if using Sedona Workbench to make a direct Sox connection to a device, app save (get) and backup locations are local to your Workbench PC. In this case, you can use the standard Directory Chooser to change to another local store or source location for the saved app or backup.

About the Sedona Environment Manager

In TXS 1.2, to use the provisioning tools through a station you must first make a platform connection and use Sedona Environment Manager to install and configure the Sedona environment under the JACE. For details on the Sedona Environment Manager, refer to the *NiagaraAX Sedona Framework Networks Guide*.

About the Sedona Tools and Schema

The following topics are covered in this section:

- [Accessing the Sedona Tools](#)
- [Sedona Tools overview](#)
- [Schema overview](#)

Accessing the Sedona Tools

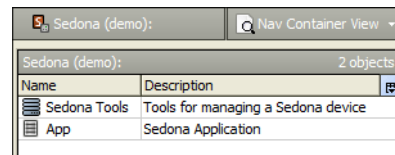
The Sedona Tools can be accessed using either of the connection methods described here:

- [For a Sox session connection](#)
- [For a Fox connection to a station with a networked device](#)

For a Sox session connection

To provision a device through a Sox session connection (either direct or tunneled), the initial Sox connection shows the Sedona Tools container and App component in a Nav Container view, as shown in [Figure 1-2](#). The Sedona Tools View is the container for the three provisioning tools (Kit Manager, Application Manager, and Backup/Restore Tool). You can double-click Sedona Tools to open the Sedona Tools view where you can select an individual provisioning tool to use.

Figure 1-2 Initial view after making Sox connection



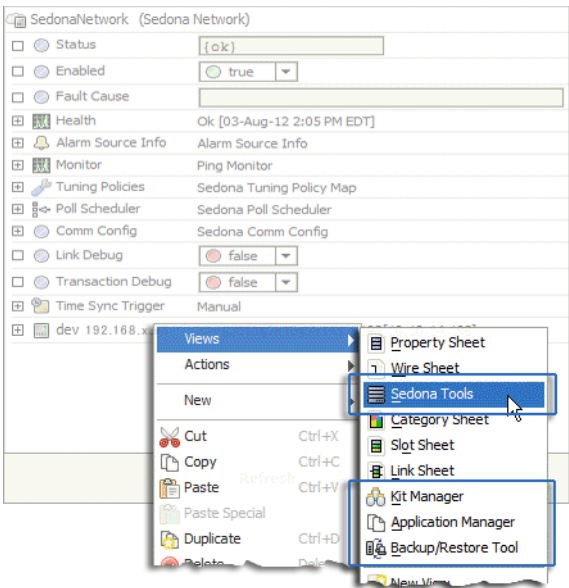
For a Fox connection to a station with a networked device

To provision a Sedona device via the Fox connection to a JACE station, the initial Fox connection shows the Station Summary view. In this view, you can expand the Config and Drivers nodes to locate and expand a Sedona Network. Access the Sedona Tools by right-clicking a Sedona device to see the Views dropdown list where you can select the provisioning tools, as shown in [Figure 1-3](#).

Note: The Sedona Tools can also be found under a device in the Jen6lp network.

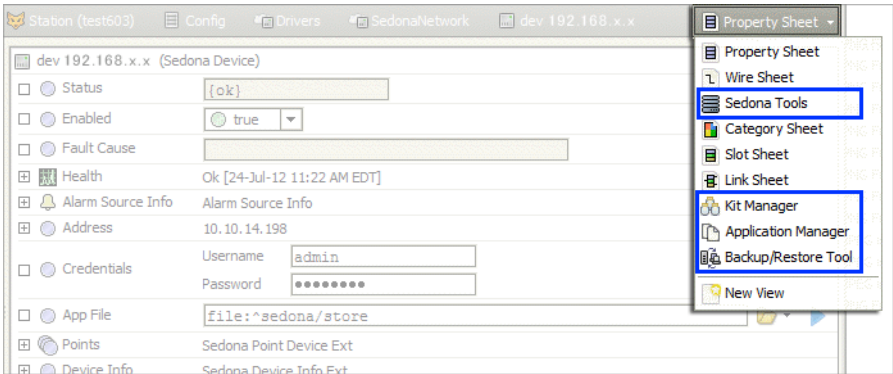
The Sedona Tools View is the container for the three provisioning tools (Kit Manager, Application Manager, and Backup/Restore Tool) that are also shown in the drop-down list.

Figure 1-3 Accessing Sedona Tools from networked device Views drop-down



Note: The provisioning tools can also be accessed by expanding the Sedona Device node in the Nav Tree, or on the Sedona Device Property Sheet view, or by clicking the Sedona Device view drop-down list in the upper right corner, as shown in [Figure 1-4](#).

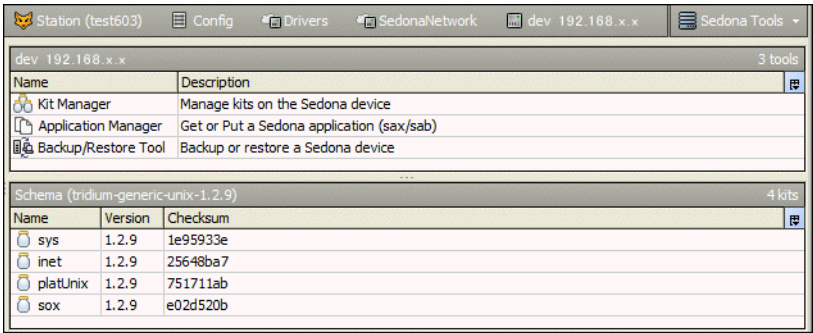
Figure 1-4 Sedona Device view drop-down list



Sedona Tools overview

The Sedona Tools view is the default view of the Sedona Tools container. This view consists of the Tools and Schema tables shown in [Figure 1-5](#). Use this view for all Sedona Framework platform provisioning of the device, including all management of applications and kits, as well as performing backup and restore operations.

Figure 1-5 Sedona Tools view

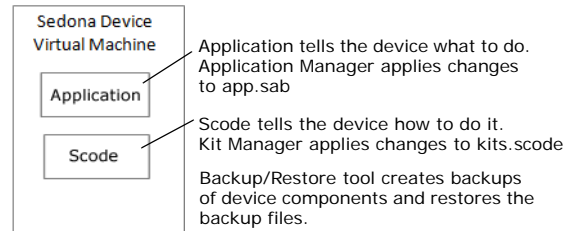


The three provisioning tools available in the Sedona Tools view are:

- **Kit Manager**
Manages the kits.scode on a Sedona Framework device. This tool is used to add and remove kits, and to change the kit versions running on a device.
- **Application Manager**
Manages applications on a Sedona Framework device. Application Manager uses the Get and Put functions to copy applications from or to a device.
- **Backup/Restore Tool**
Creates backup files for Sedona Framework device components as zip files and restores the backups. These backup files can contain the Sedona virtual machine (SVM), the app.sab and kits.scode, as well as the kits, manifests, and PAR files.

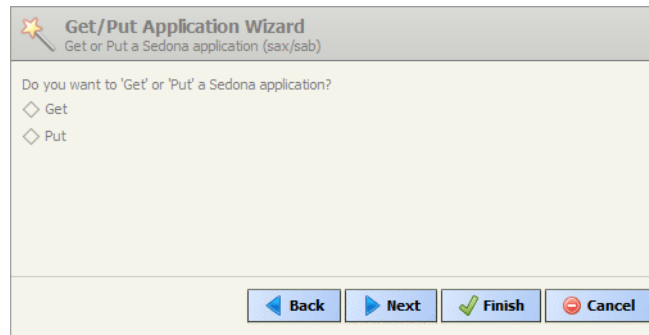
The diagram in [Figure 1-6](#) below shows the relationship of the provisioning tools to the components of the Sedona virtual machine.

Figure 1-6 Relationship of Provisioning Tools to SVM components



The Sedona Tools are wizard-type tools which simplify the provisioning process by presenting you with a sequence of dialog boxes that lead you through a series of steps. Use the controls (a set of buttons positioned at the bottom of the dialog boxes) shown in [Figure 1-7](#) to navigate through the individual steps of a procedure.

Figure 1-7 Sedona Tools controls



The buttons function as follows:

- The **Next** button advances to the next step in a procedure.
- The **Back** button returns to the previous step in a procedure.
- The **Finish** button completes a procedure.
- The **Cancel** button aborts a procedure.

Note: When a button has a dimmed appearance that indicates an unavailable navigation choice.

Schema overview

The schema is the collection of “kits” available on the device for use by the application currently running on a device. The schema can be viewed in the Schema table located in the lower portion of the Sedona Tools view, as shown in [Figure 1-8](#).

Each row of the table represents a kit. Each kit contains some number of named types. The app contains components, which are instances of a Sedona type, just as a Niagara station contains AX components, which are instances of Niagara types. Each type in a kit declares zero or more slots. For any given version of a kit, there is a fixed list of types and their declared slots. The checksum for each kit is generated when the Sedona Framework compiler creates a kit file. The specific list of kit parts (kits at a specific checksum revision) is called a schema.

Figure 1-8 Schema table

The screenshot shows the Sedona Tools application interface. At the top, there is a menu bar with options: Station (test603), Config, Drivers, SedonaNetwork, dev 192.168.x.x, and Sedona Tools. Below the menu bar, there are two main sections. The first section is titled 'dev 192.168.x.x' and contains a table with 3 tools. The second section is titled 'Schema (tridium-generic-unix-1.2.9)' and contains a table with 4 kits.

Name	Description
Kit Manager	Manage kits on the Sedona device
Application Manager	Get or Put a Sedona application (sax/sab)
Backup/Restore Tool	Backup or restore a Sedona device

Name	Version	Checksum
sys	1.2.9	1e95933e
inet	1.2.9	25648ba7
platUnix	1.2.9	751711ab
sox	1.2.9	e02d520b

CHAPTER 2

About The Kit Manager

The Kit Manager tool allows you to add and remove kits, as well as to change version levels of kits deployed in the scode running on the device. Also, you have the option to force the kits.scode to be rebuilt.

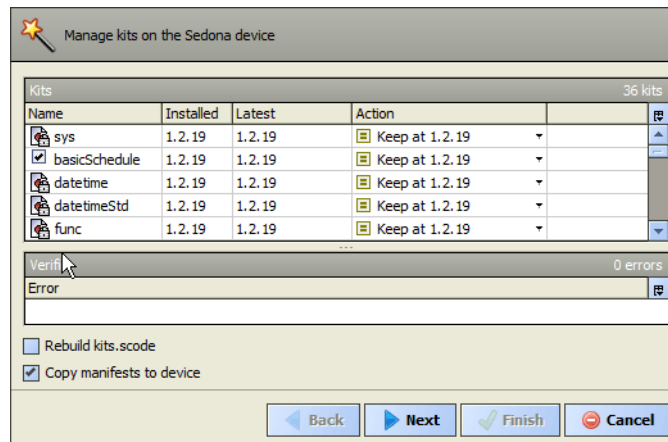
The appropriate kits must be available in the Sedona environment you are using in order to change the kits on a device using the Kit Manager. All of the open source kits and Niagara-specific kits are provided in the TXS bundle installation. Third party vendors provide their own kits and instructions for downloading and installing them.

Note: *On the JACE-2 series platform only, the “maxHeap” license feature is required in order to compile scode. Provisioning attempts in Kit Manager will fail without this feature. For more details on this, refer to the Troubleshooting section, “Provisioning fails when compiling scode on JACE-2 series platform” on page A-2.*

About the Kit Manager view

When launched, Kit Manager analyzes the kits installed on a device. A progress bar displays in the view during this analysis. Afterwards, the Kit Manager view displays featuring the Kits table and Verifier table, as shown in [Figure 2-1](#).

Figure 2-1 Kit Manager view



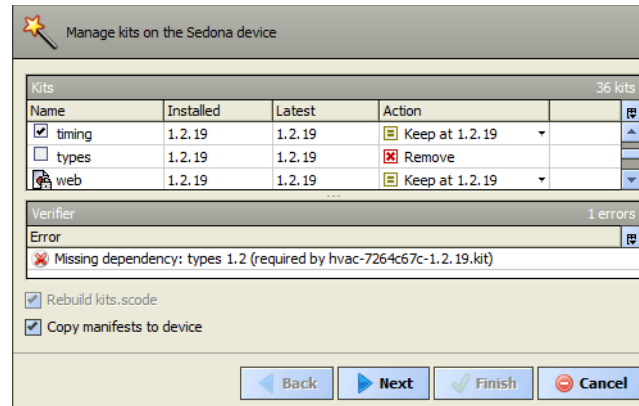
The Kits table shows the schema or collection of “kits” used by the application currently running on the device. Each row in the Kits table represents a kit. Also shown for each kit, is the specific version installed on the device, the latest version downloaded to either the Workbench (if managing the device through a Sox session) or to the JACE (if managing the device through the Sedona Device in the Sedona Network), as well as the individual actions available for the kit.

In the Kits table, a kit icon with a padlock (🔒) shown to the left of the kit name indicates that the kit is required by the application running on the device and may not be deleted. An *unchecked* box to the left of a kit name indicates that the kit is compatible and may be installed. A *checked* box indicates a kit that is installed, but not used by the current Sedona app. Kits not used by the current app may be uninstalled, if desired. Kits that are not supported by the device appear dimmed on a gray background. The Action column describes those kits as “Not supported”.

The Verifier table shows kit dependency errors when they occur. If a selected kit requires the presence of an uninstalled kit in order to run on the device, the missing dependency will be listed in the Verifier table, as shown in [Figure 2-2](#).

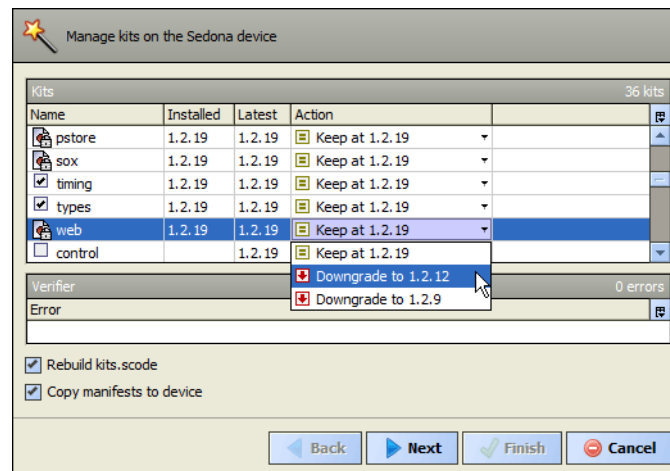
Note: When a kit dependency error occurs, the Kit Manager will NOT automatically select a missing kit (even if it is present in the Sedona environment and available for installation) since doing so may cause kits to be deployed to the device that the user did not intend. The missing kit, if not present in the Workbench, must be obtained from the vendor.

Figure 2-2 Verifier table shows that removing the types kit causes a missing dependency for the hvac kit.



The actions that are available for each installed kit can be viewed in the **Action** drop-down list which you access by clicking a kit's Action value, as shown in [Figure 2-3](#).

Figure 2-3 Kit Manager dialog box



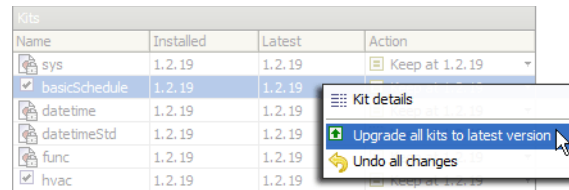
The **Action** drop-down list options can be one or more of the following:

- **Keep at** [current version]
- **Upgrade to** [newer version]
- **Downgrade to** [older version]

The newer/older kit version levels shown in the **Action** drop-down menu are available in the Sedona environment. They are stored in the `SEDONAHOME\kits` directory.

If you have older kits in the current app, you may want to upgrade all kits to the newer versions available in your Workbench. You can easily do this by right-clicking within the Kits table to display the drop-down list shown in [Figure 2-4](#), and then select the option to **Upgrade all kits to latest version**.

Figure 2-4 Right-click in the Kits table to display the Kit details drop-down list



The **Kits table** drop-down list options function as follows:

- **Kit details**
This option opens a pop-up window listing details for the selected kit. The details include kit name, current version with a drop-down list for other versions (if available), kit checksum, as well as listing any kit dependencies.
- **Upgrade all kits to latest version**
This option changes the Action value for all installed kits to Upgrade to (latest version)
- **Undo all changes**
This option returns the kits back to the unmodified state.

Located below the two tables in the Kit Manager dialog box, are check boxes for the following options:

- **Rebuild kits.scode**
This option is automatically selected (and dimmed) when you make any kit selection changes. The reason for that is that the kits.scode must be rebuilt whenever a change affects the schema. This ensures such changes are included in the recompiled binary .scode image. The only time you would need to actually select the option is when you want to force the scode to be rebuilt (even if no change was made to kit selections). This is useful in a development environment where the actual code in a kit might change while the kit version and checksum do not. In this case, no change is required in the Kit Manager but the scode must be regenerated in order to deploy the new code on a device.
- **Copy manifests to device**
This optional feature gives the device the capacity to act as a “manifest server.” If not supported by the device, the option will not display in the view. If supported by the device, when you encounter a missing manifest while copying kits, the device can serve up the missing manifest. By contrast, in Sedona 1.1 and earlier, a missing manifest causes an exception.
***Note:** Functioning as a manifest server is an optional feature that must be supported by the device in order for the option to display in the view. If you do not see the option, then it is not supported by the device. Also, the device must be running Sedona Framework TXS 1.2 in order to function as a manifest server.*

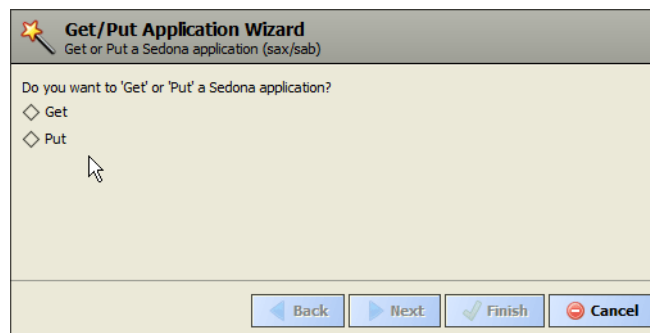
For more details on using the Kit Manager, refer to “[Common Tasks](#)” on page 5-1.

CHAPTER 3

About The Application Manager

The Application Manager tool is used to copy an application *from* using the Get operation or *to* a Sedona Device using the Put operation. When launched, Application Manager displays the Get/Put Application wizard, shown in Figure 3-1, which leads you through the steps of each procedure.

Figure 3-1 The Application Manager tool



Some reasons for using the **Get** option are:

- **To modify the application.**
You may want to make application modifications offline.
- **To save the application as a backup.**
Although, it may be more useful to make a complete backup using the Backup/Restore Tool.
- **To distribute the application.**
You may want to distribute the application to another device.

The **Put** option is used simply to replace the application that is running on a device.

Additionally, when the Get or Put is done from the context of a station, this associates the device with the app file (setting the appFile ord in the Sedona Device) which allows for offline point discovery.

About the Get and Put operations

When using the Get and Put operations, the *save* (Get) and *source* (Put) locations are determined automatically by the software and they differ slightly depending on the method used to connect to the device. You can use either of the connection methods described here:

- **For a Sox session connection**
When provisioning through Sedona Workbench or Niagara Workbench via a direct Sox connection (or through Niagara Workbench via a tunneled-Sox connection), the save/source location is:
 - local|file:!sedona/store/apps

Note: You can make a “direct Sox connection” to a Sedona Device in either Niagara Workbench or Sedona Workbench by selecting **File > Open Device**. You can make a “tunneled Sox connection” through the JACE only in Niagara Workbench by right-clicking a Sedona Device in an opened station and selecting **Open Sox-Tunnel Session** (this requires the SoxTunnelService to be in the station’s Services folder).

Also, you can change the save/source location *only* when provisioning through a Sox session connection (direct Sox or Sox-tunneled). In this case, you can use either of the controls on the right side of the dialog box to select another save/source location.

- **For a Fox connection to a JACE station**

When provisioning a networked device via a Fox connection to a JACE station the save/source location is:

- `ip:<IpAddress>|fox:|file:^sedona/store/<SedonaNetwork_Name>/<SedonaDevice_Name>/apps`

Note: The terms “*IpAddress*”, “*SedonaNetwork_Name*” and “*SedonaDevice_Name*” shown here are not fixed values, they are placeholders for the actual IP number of your JACE as well as whatever you have named your Sedona Network and Sedona Device.

You **cannot** change the save/source location if you are provisioning via a networked device on the JACE station. The save/source location is the /apps folder in the dedicated file space for the connected device on the JACE. However, you can transfer files to this location and manage files there by using the available controls at the far right side of the dialog box to open a Directory view of the /apps folder. For more details, refer to *Niagara Users Guide*.

For more details on using the Get and Put operations, refer to “[Common Tasks](#)” on page 5-1.

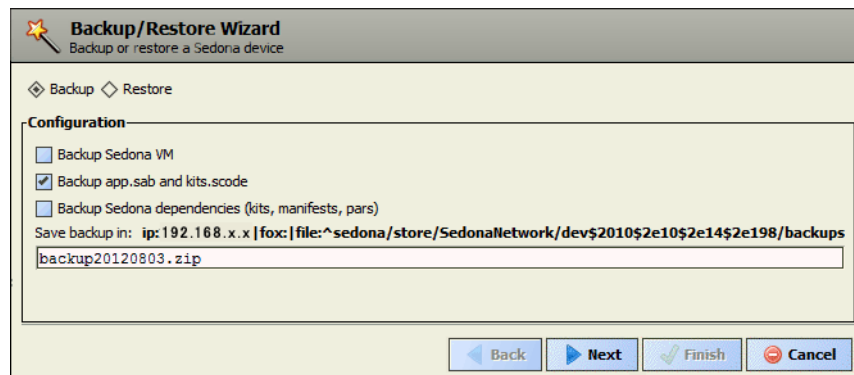
CHAPTER 4

About The Backup/Restore Tool

As the name implies, the Backup/Restore Tool is used to backup and restore Sedona Framework device components. By default, the Backup option is the initial view of the Backup/Restore Tool, as shown in [Figure 4-1](#). You can change to the Restore option by clicking on **Restore** located just above the configuration area, on the left side of the dialog box.

When creating a backup of a device, the data is saved in a zip file. When restoring a backup to a device, the contents of the selected zip file are saved to the connected device.

Figure 4-1 The Backup/Restore Tool dialog box (provisioning through networked station)



About the Backup and Restore operations

A backup file can be configured to contain the Sedona virtual machine (SVM), the app.sab and kits.scodes, as well as all of the Sedona dependencies (kits, manifests, and par files). The latter is one of the major enhancements in Sedona Framework TXS 1.2. This added ability of the Niagara station backup to include Sedona Framework device files, ensures the presence of all of the files that your application requires to run if deployed on another device. In this sense, the backup tool provides a true “backup”. Once you have created a backup of a device, if the device fails you can quickly get operations running again by restoring the backup to a replacement device.

Three backup options are available:

- **Backup Sedona VM**
This option backs-up the virtual machine.
Note: This option is device-specific. If not supported by your device, selecting the option will cause the backup operation to fail. Refer to your device documentation for information on backing up the virtual machine.
- **Backup app.sab and kits.scodes**
This option is selected by default. It provides a minimal backup containing the app.sab and kits.scodes.
- **Backup Sedona dependencies (kits, manifests, pars)**
This option includes all additional files needed for the Workbench or JACE to connect to and provision the device.
Note: This option requires the backup of the app.sab and kits.scodes as well.

When using the Backup and Restore operations, the *save* (backup) and *source* (restore) locations differ slightly depending on the method used to connect to the device. You can use either of the connection methods described here:

- **For a Sox session connection**

When provisioning through Sedona Workbench or Niagara Workbench via a direct Sox connection (or through Niagara Workbench via a tunneled-Sox connection), the save/source location is:

- `local|file:!sedona/store/backups`

Note: You can make a “direct Sox connection” to a Sedona Device in either Niagara Workbench or Sedona Workbench by selecting **File > Open Device**. You can make a “tunneled Sox connection” through the JACE only in Niagara Workbench by right-clicking a Sedona Device in an opened station and selecting **Open Sox-Tunnel Session** (this requires the SoxTunnelService in the station’s Services folder).

You can change the save/source location *only* when provisioning through a Sox session connection (direct Sox or Sox-tunneled). In this case, you can use either of the controls on the right side of the dialog box to select another save/source location.

- **For a Fox connection to a JACE station**

When provisioning a networked device via a Fox connection to a JACE station the save/source location is:

- `ip:<IpAddress>|fox:|file:^sedona/store/<SedonaNetwork_Name>/<SedonaDevice_Name>/backups`

Note: The terms “IpAddress”, “SedonaNetwork_Name” and “SedonaDevice_Name” shown here are not fixed values, they are placeholders for the actual IP number of your JACE as well as whatever you have named your Sedona Network and Sedona Device.

You **cannot** change the save/source location if you are provisioning via a networked device on the JACE station. The save/source location is the /backups folder in the dedicated file space for the connected device on the JACE. However, you can transfer files to this location and manage files there by using the available controls at the far right side of the dialog box to open a Directory view of the /backups folder. For more details, refer to *Niagara Users Guide*.

For more details on using the Backup and Restore operations, refer to “[Common Tasks](#)” on page 5-1.

CHAPTER 5

Common Tasks

The Sedona Tools are used to provision a single Sedona Framework device.

Common Sedona provisioning tasks include the following:

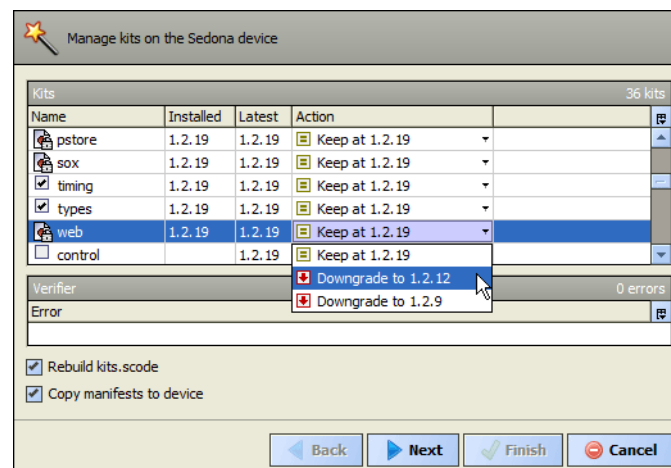
- [Make kit selection changes](#)
- [Get the app from a device](#)
- [Put an application on a device](#)
- [Create a backup](#)
- [Restore a backup](#)

Make kit selection changes

This procedure describes how to make kit selection changes (add/remove kits or change kit version levels) to kits deployed on a Sedona Device.

- Step 1 Access the Sedona Tools under a Sedona Device:
- **For a Fox connection to a station with a networked device**
When provisioning a networked device through the JACE station, expand the Sedona Network, right-click the device and select **Views** menu where you can access the provisioning tools.
 - **For a Sox session connection (either direct or Sox-tunneled)**
When provisioning a device through a Sox session connection (either direct or Sox-tunneled), double-click the **Sedona Tools** container to access the provisioning tools.
- Step 2 Select the **Kit Manager** to launch the Kit Manager wizard. After analyzing the kits currently installed on the device, the Kit Manager view is displayed.
- Step 3 In the Kits table, make selections as needed to add or remove kits, and to change kit versions.
- Add a kit by clicking on the kit's unchecked check box.
 - Remove an installed kit by clicking on the kit's checked check box to clear it.
 - Change a kit version by clicking on the kit's Action value, as shown in [Figure 5-1](#).

Figure 5-1 Kit Manager, showing Action menu for a kit



The **Action** drop-down list options can be one or more of the following:

- **Keep at** [current version]
- **Upgrade to** [newer version]
- **Downgrade to** [older version]

Note: The newer/older kit version levels shown in the **Action** drop-down menu reflect what is available in the Sedona environment. They are stored in the `SEDONAHOME/kits` directory

Step 4 Check the Verifier table for kit dependency errors resulting from the your selections.

Note: If a selected kit requires the presence of an uninstalled kit in order to run on the device, the missing dependency will be listed in the Verifier table. Also, if a kit dependency error occurs as a result of your selections, the Kit Manager will NOT automatically select a missing kit. If the missing kit is not present in Workbench, you can obtain it from the vendor.

Step 5 If desired, click the check box to **Rebuild kits.scode**.

Note: The option to “Rebuild kits.scode” is selected automatically whenever you change kit selections and/or change kit versions since the schema will need to be updated.

Step 6 If desired, click the check box to **Copy manifests to device**.

Note: Functioning as a manifest server is an optional feature that must be supported by the device in order for the option to display in the view. If you do not see the option, then it is not supported by the device.

Step 7 Click the **Next** button. The tasks to be performed are listed in the upper portion of the view.

Step 8 Click the **Finish** button to execute the tasks. As each task is completed, the status (Success or Failed) is displayed to the right of the task. Also, a log can be viewed in the pane. On successful completion you are prompted to restart the device.

Step 9 Click the **Restart Device** button.

Note: Restarting the device will disconnect your connection. To re-open your Sox connection, double-click the dimmed Sedona node in the Nav tree, and enter the log-in credentials.

For more information, refer to “[About The Kit Manager](#)” on page 2-1.

Get the app from a device

This procedure describes how to use the Get operation to copy the app from a Sedona Device which saves the .sax file to the app database in the Sedona environment you are using.

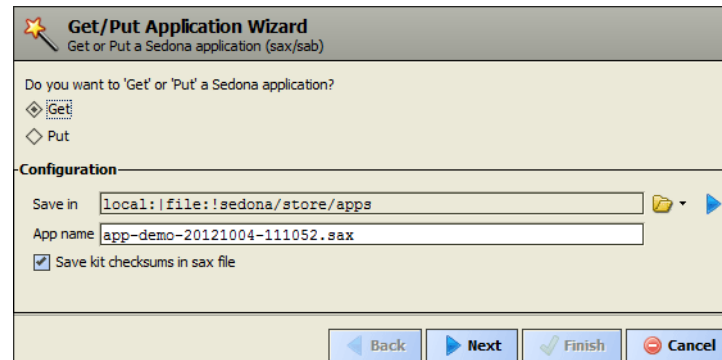
Step 1 Access the Sedona Tools under a Sedona Device:

- **For a Fox connection to a station with a networked device**
When provisioning a networked device through the JACE station, expand the Sedona Network, right-click the device and select **Views** menu where you can access the provisioning tools.
- **For a Sox session connection (either direct or Sox-tunneled)**
When provisioning a device through a Sox session connection (either direct or Sox-tunneled), double-click the **Sedona Tools** container to access the provisioning tools.

Step 2 Select the **Application Manager** to launch the Get/Put Application wizard.

Step 3 Click the **Get** option. This populates the configuration area of the dialog box with a save location and default app filename, and the selected option to include kit checksums in the file, as shown in [Figure 5-2](#).

Figure 5-2 Get dialog box (for a Sox session connection)



The **Save in** location for the app is the /apps folder in the dedicated file space for the connected device in the Sedona environment you are using.

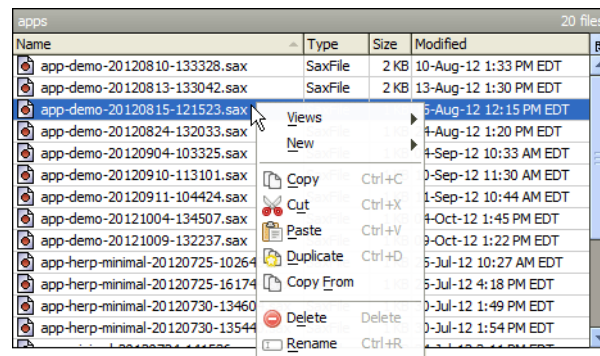
Note: The “Save in” location is determined automatically by the software and it differs slightly depending on whether you are provisioning through the Workbench via a Sox session connection or via a Fox connection to a networked device on the JACE station. For details, refer to “About The Application Manager” on page 3-1.

The Save in location can be modified **only** when making a Sox session connection (either direct Sox or Sox-tunneled). In this case, you can click the Directory icon to launch a Directory Chooser and browse to select an alternate save location. If you are provisioning through a Fox connection to a networked device on the JACE station, the **Save in** location **cannot** be modified.

With either connection method described above, you can use the Directory view to manage files in the /apps folder. Click the blue triangle icon positioned at the far right side of the Get dialog box to open the Directory view. The Directory view allows you to do the following:

- Copy app files into the /apps folder using the drag-and-drop method
- Right-click in the Directory view to display other file management options, as shown in [Figure 5-3](#):
 - Create New folders and move app files for further organization
 - Cut, Copy, Paste, and Duplicate app files
 - Copy From another location
 - Move app files among folders
 - Rename app files
 - Delete app files

Figure 5-3 Right-click in the Directory view to manage files in /apps folder



Step 4 Use the default application file name or modify it as needed. By default the **App name** of app-deviceName-applicationName-YYYYMMDD-HHHHSS.sax is used, where YYYY is year, MM is month, DD is day, HHHH is hours, SS is seconds. You can change the app filename as needed.

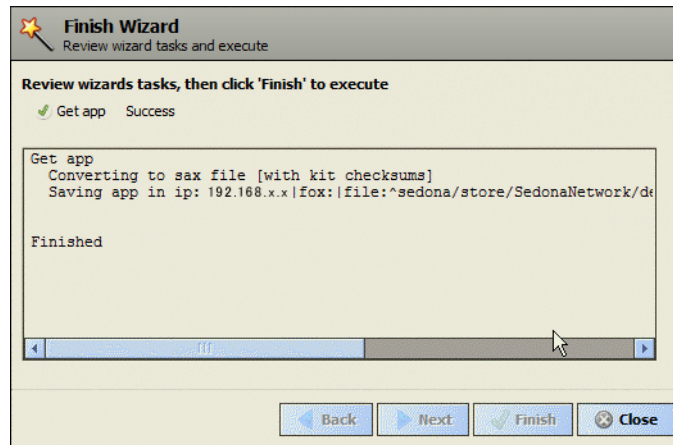
Step 5 Use the default selection to **Save kit checksums in sax file** or deselect it as needed.

Note: If checksums are saved, then the sax file will only run with those exact versions of kits. If checksums are not saved, the sax file can be installed onto a device with the required kits without regard to their checksum. It is a safer practice, though more restrictive, to save the checksums in the app. However, if you are regularly making changes to the kits, for example during development, then you may want to uncheck this option..

Step 6 Click the **Next** button. The tasks to be performed will be listed in the upper portion of the screen.

Step 7 Click the **Finish** button to execute the tasks. As each task is completed, the status (Success or Failed) is displayed to the right of the task. Also, a log can be viewed in the pane. On completion you are notified that provisioning has finished, as shown in [Figure 5-4](#).

Figure 5-4 Completed Get option (Fox connection to a station on the JACE)



Step 8 Click the **Close** button to exit the Get/Put Application wizard.

Put an application on a device

This procedure describes how to use the Put operation to copy an app from the app database in the Sedona environment to a Sedona Device.

Step 1 Access the Sedona Tools under a Sedona Device:

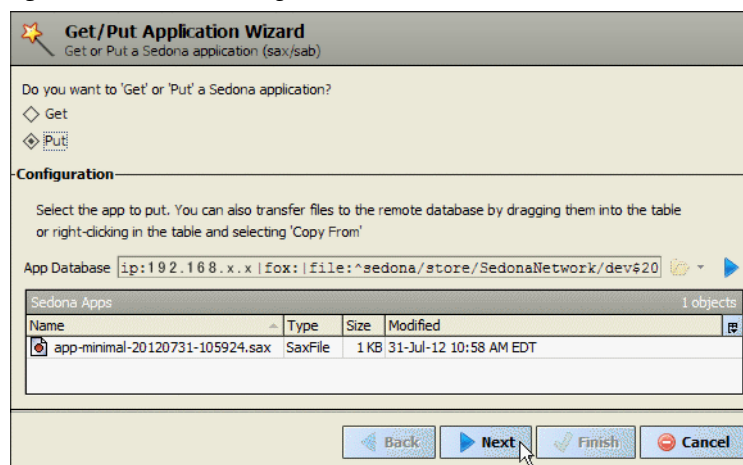
- **For a Fox connection to a station with a networked device**
When provisioning a networked device through the JACE station, expand the Sedona Network, right-click the device and select **Views** menu where you can access the provisioning tools.
- **For a Sox session connection (either direct or Sox-tunneled)**
When provisioning a device through a Sox session connection (either direct or Sox-tunneled), double-click the **Sedona Tools** container to access the provisioning tools.

Step 2 Select the **Application Manager** to launch the Get/Put Application wizard.

Step 3 Select the **Put** option. This populates the configuration area of the dialog box with the **App Database** source location, and the Sedona Apps table listing the applications stored there. You are prompted to select an app to be copied to a Sedona device, as shown in [Figure 5-5](#).

Note: On the JACE-2 series platform only, the “maxHeap” license feature is required in order to compile scode. The Put function will fail without this feature. For more details on this, refer to the Troubleshooting section, “[Provisioning fails when compiling scode on JACE-2 series platform](#)” on page A-2.

Figure 5-5 The Put dialog box (Fox connection to a station on the JACE)



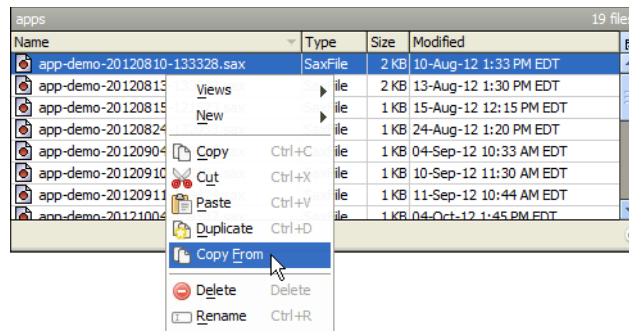
Note: The **App Database** source location is determined automatically by the software and it differs slightly depending on whether you are provisioning through the Workbench via a Sox session connection or via a Fox connection to a networked device on the JACE station. For details, refer to “[About The Application Manager](#)” on page 3-1.

The source location can be modified **only** when making a Sox session connection (either direct Sox or Sox-tunneled). In this case, you can click the Directory icon to launch a Directory Chooser and browse to select an alternate source location. If you are provisioning through a Fox connection to a networked device on the JACE station, the source location **cannot** be modified.

With either connection method described above, you can always transfer files to the source location and manage files there by displaying a Directory view of the /apps folder. Click the blue triangle icon positioned at the far right side of the Put dialog box to open the Directory view. The Directory view allows you to manage files in the /apps folder and copy apps from another location as described below:

- Copy app files into the Directory view using the drag-and-drop method
- Right-click in the Directory view to display other file management options, as shown in Figure 5-6:
 - Create New folders and move app files for further organization
 - Cut, Copy, Paste, and Duplicate app files
 - Copy app files from another location using the **Copy From** option
 - Delete app files
 - Rename app files

Figure 5-6 Right-click in the Directory view to manage files in /apps folder



Step 4 In the **Sedona Apps** table, select the application file name to be copied to the device.

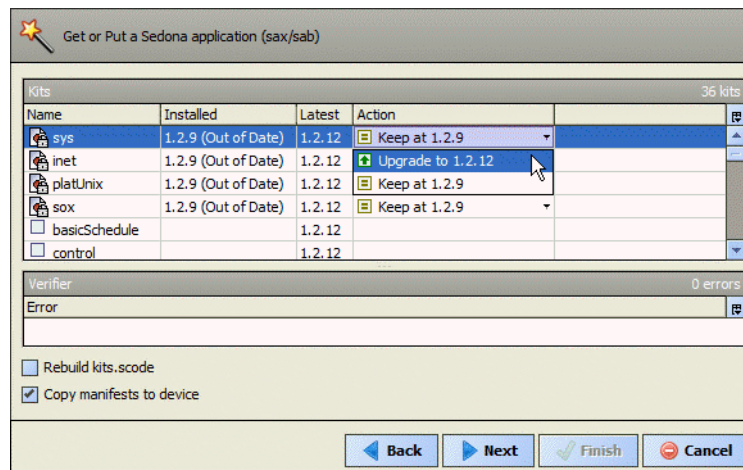
Step 5 Click the **Next** button. This launches the Kit Manager which ensures that the appropriate kits are deployed on the device in order to support the app.

In the Kit Manager view, you can change the version level of any installed kit by clicking on the individual Action value to reveal the **Action** drop-down list, as shown in Figure 5-7.

Note: Actions in the drop-down list are based on the version levels available in the local Workbench.
Possible actions are:

- Downgrade to (earlier version)
- Keep at (current version)
- Upgrade to (newer version)

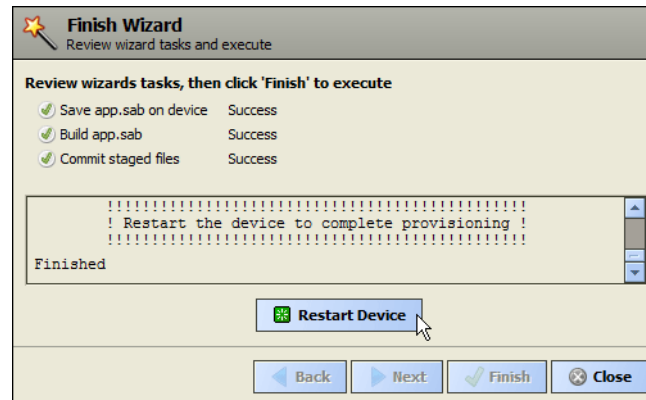
Figure 5-7 Kit Manager with Action drop-down list showing two available version levels for the sys kit



Step 6 Click the **Next** button. The tasks to be performed will be listed in the upper portion of the screen.

- Step 7 Click the **Finish** button to execute the tasks. As each task is completed, the status (Success or Failed) is displayed to the right of the task. Also, a log can be viewed in the pane. On completion you are notified that provisioning has finished, as shown in [Figure 5-8](#).

Figure 5-8 Put option completed successfully



After the Put operation completes successfully, you must restart the device.

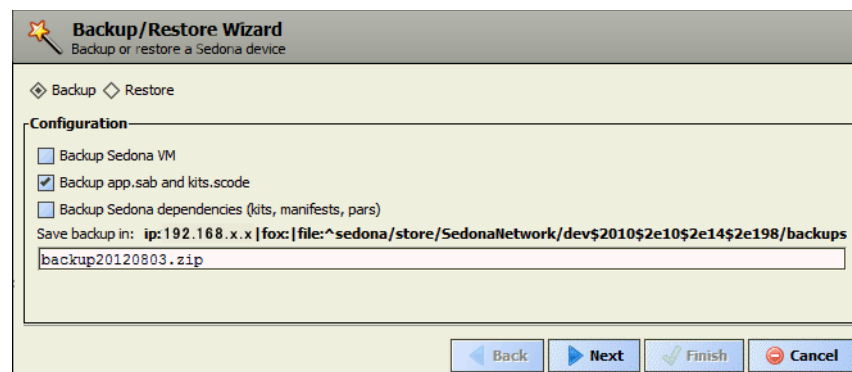
Note: Restarting the device will disconnect your connection. To re-open your Sox connection, double-click the Sedona node in the Nav tree, and enter the log-in credentials.

Create a backup

This procedure describes how to create a zip file backup of the selected items on a Sedona Device and saves the zip file to the Sedona environment.

- Step 1 Access the Sedona Tools under a Sedona Device:
- **For a Fox connection to a station with a networked device**
When provisioning a networked device through the JACE station, expand the Sedona Network, right-click the device and select **Views** menu where you can access the provisioning tools.
 - **For a Sox session connection (either direct or Sox-tunneled)**
When provisioning a device through a Sox session connection (either direct or Sox-tunneled), double-click the **Sedona Tools** container to access the provisioning tools.
- Step 2 Select the **Backup/Restore Tool** to launch the Backup/Restore wizard. By default, the **Backup** option is selected and a default backup filename, backupYYYYMMDD.zip, is used. Where YYYY is year, MM is month, DD is day, as shown in [Figure 5-9](#). You can modify the backup filename as needed.
- Note:** The save location is determined by the software and differs slightly depending on the method used to connect to the device. For more information, refer to [“About the Backup and Restore operations”](#) on page 4-1

Figure 5-9 The Backup/Restore Tool dialog box (using a Fox connection to a JACE station)

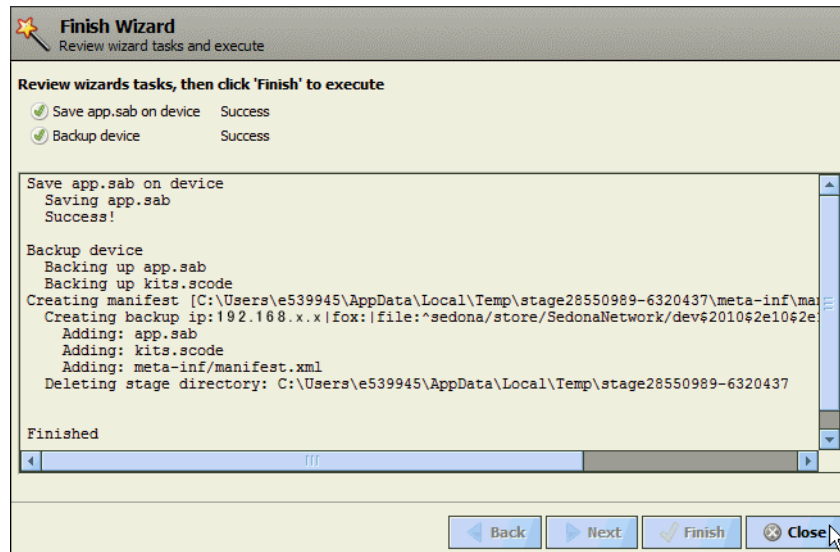


- Step 3 Click configuration check boxes to select the device components to be backed-up to the local Sedona environment, and use the default backup file name or you can modify it as needed.

Note: The first configuration option, “Backup Sedona VM”, is device-specific. If not supported by your device, selecting it will cause the backup operation to fail. Refer to your device documentation for information on creating a backup of the virtual machine.

- Step 4 Click the **Next** button. The tasks to be performed will be listed in the upper portion of the screen.
- Step 5 Click the **Finish** button to execute the tasks. As each task is completed, the status (Success or Failed) is displayed to the right of the task. Also, a log can be viewed in the pane. On completion you are notified that provisioning has finished, as shown in [Figure 5-10](#).

Figure 5-10 Backup function completed successfully (using a Fox connection to a JACE station)



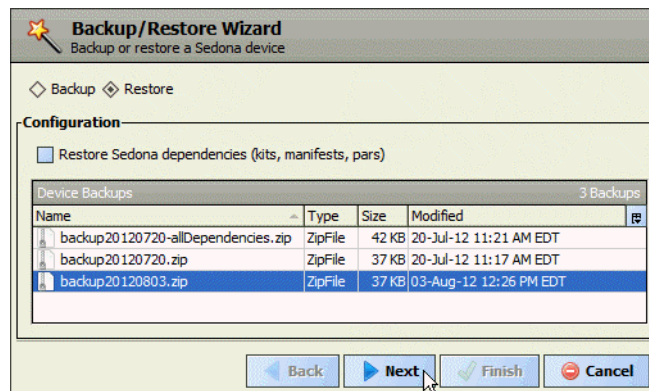
- Step 6 Click the **Close** button to exit the Backup/Restore Wizard.

Restore a backup

This procedure describes how to restore a zip file backup by installing the contents of the selected zip file to a Sedona Device.

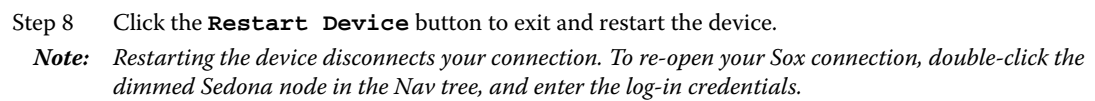
- Step 1 Access the Sedona Tools under a Sedona Device:
- **For a Fox connection to a station with a networked device**
When provisioning a networked device through the JACE station, expand the Sedona Network, right-click the device and select **Views** menu where you can access the provisioning tools.
 - **For a Sox session connection (either direct or Sox-tunneled)**
When provisioning a device through a Sox session connection (either direct or Sox-tunneled), double-click the **Sedona Tools** container to access the provisioning tools.
- Step 2 Select the **Backup/Restore Tool** to launch the Backup/Restore wizard.
- Step 3 Click the **Restore** option.
- Step 4 In the **Device Backups** table, select a backup file to install on the device, as shown in [Figure 5-11](#).

Figure 5-11 Restore dialog box with default selection



- Step 5 Click the check box to **Restore Sedona dependencies (kits, manifests, pars)** if you want to restore those. They will be installed, overwriting the kits, manifests, and pars on the device.
- Step 6 Click the **Next** button. The tasks to be performed will be listed in the upper portion of the dialog box.
- Step 7 Click the **Finish** button to execute the tasks.

Figure 5-12 *Restore function completed successfully*



FAQs and Troubleshooting

Frequently asked questions about the provisioning tools and troubleshooting tips for problems that you may experience while provisioning a Sedona Framework device are covered in this section:

- [Frequently asked questions](#)
- [Troubleshooting tips](#)

Problems not listed here may be network issues. For information on Sedona Framework networks, refer to the *NiagaraAX Sedona Framework Networks Guide*.

Frequently asked questions

Below are a few frequently asked questions (FAQs) about the Sox Tools.

Q: Is there any difference between the Sox tools that are available in Sedona Framework TXS-enabled Niagara Workbench and those available in Sedona Workbench?

A: The same provisioning tools are available and the tools function the same in both versions of Workbench. You will notice a difference when you are provisioning through a Sedona network on a JACE, see the discussion on, [“Changes to provisioning tools for TXS 1.2”](#).

Note: *To use the Sedona tools under a networked Sedona Device on a JACE station, you must first configure the JACE controller via a platform connection, using the “Sedona Environment Manager” view.*

Q: What is a PAR file?

A: The Platform Archive (PAR) file is a zip file with a .par extension. The PAR provides a way of distributing various files and metadata about a platform into a single entity. Basically, it describes the platform. For more information on platform archive, refer to the [“Sedona Tools terms”](#) in the Preface section.

Q: Is there a relationship between the schema of a Sedona Framework device and its platform archive, and if so, what?

A: No, the schema is completely independent of the platform archive. For more information on schema and platform archive, refer to the [“Sedona Tools terms”](#) in the Preface section.

Q: How can I avoid the situation where a kit fails to install on a Sedona Framework device due to insufficient space?

A: Currently, there is no way to avoid this. The maximum file size is device-dependent. Sedona does not provide a way to know a device’s maximum file size.

Q: Why do certain provisioning operations fail on my JACE-2 series controller?

A: On the *JACE-2 series*, the maxHeap license feature is required in order to compile scode. Provisioning operations that involve compiling the scode will fail without the maxHeap feature. For more details on this, refer to, [“Provisioning fails when compiling scode on JACE-2 series platform”](#) on page A-2.

Q: On my JACE with a TXS 1.2 Sedona Network, would I ever need to open a Sox session connection to a device directly or make a Sox-tunnel connection to a device?

A: A Sox connection would be necessary if you are using Sedona Workbench or if you have a standalone Sedona device with no Niagara integration. Also, it may also be useful for testing and troubleshooting purposes to open a Sox session connection to a device (either direct Sox or Sox-tunneled). Using a Sox connection, scode and sax files compile faster on your PC compared to on the JACE. You may find other ways in which performance is better with a Sox connection.

Note: *Copying/pasting from one app to another where the schemas are different could cause unexpected behavior. In cases of schema mismatch, the Workbench connected via a Sox session protects against potential problems by not allowing the paste operation, whereas the Sox Gateway via the Workbench connected to the Niagara Station on the JACE does not.*

Troubleshooting tips

The following troubleshooting tips are described below:

- [Adding a kit operation fails](#)
- [Kit manager fails to load](#)
- [Provisioning fails during backup](#)
- [Provisioning fails when compiling scode on JACE-2 series platform](#)

- Provisioning fails when Sedona Environment not configured on platform

Adding a kit operation fails

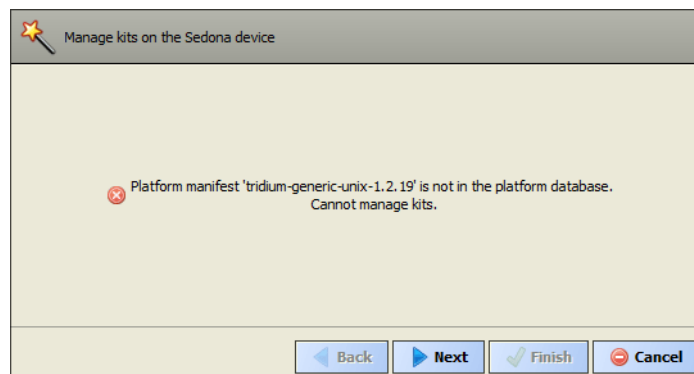
This can occur if the kit you are attempting to add exceeds the maximum file size for the device. Refer to the vendor's documentation for information on the maximum file size for the device.

Kit manager fails to load

Navigating to the Sedona Tools view reveals a warning about a missing platform manifest. When you attempt to launch the Kit Manager it fails to load. The following error message displays, “Platform manifest ‘<FileName>’ is not in the platform database. Cannot manage kits”, as shown in [Figure A-1](#).

Note: The term “FileName” shown in the preceding paragraph is not a fixed value, it is a placeholder for the name of the missing platform manifest that is referenced in the error message.

Figure A-1 Kit Manager error message for missing PAR



The error message indicates that the referenced platform manifest for the connected Sedona device is missing from the platform database. A platform manifest is included in a platform archive (PAR). The PAR is required by the Tools. It describes a platform, similar to the way a manifest describes a kit. Vendors should include a PAR as part of their build. For more information on PAR files, see [sedonadev.org \(http://sedonadev.org/doc/par.html\)](http://sedonadev.org/doc/par.html).

This error is caused by either of the following situations:

- One or more manifests are missing from the platform database in the Sedona environment being used.
- If provisioning through the JACE, there may not be any platform database at all--because a Sedona environment has not been installed on the JACE.

To correct the problem, you need to make a platform connection to the JACE and use the Sedona Environment Manager tool to install the environment files (kits/manifests/PARs) that you need for the Sedona devices on your system. For more information, the "Sedona environment management" section in the, *NiagaraAX Sedona Framework Networks Guide*.

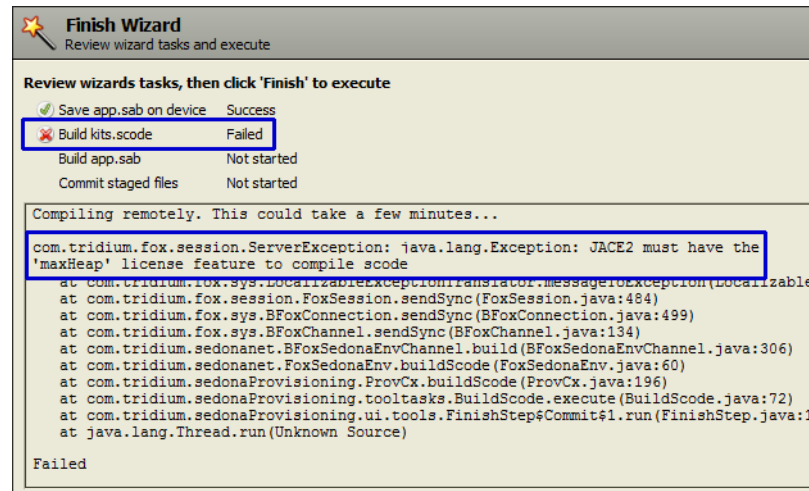
Provisioning fails during backup

The complete error message states: “sedona.sox.SoxException: Request failed: cannot open” followed by “Provisioning Failed”. This error occurs when attempting to backup the Sedona VM on a device that does not support it. The option to backup the Sedona VM is device-specific. Certain devices support it while others do not. To correct the problem, refer to the vendor’s documentation for information on making a backup of the virtual machine.

Provisioning fails when compiling scode on JACE-2 series platform

On the **JACE-2 series platform only**, provisioning operations in the Kit Manager and in the Put function in the Application Manager fail to build the scode. In both situations, after you press **Finish**, the tool appears to begin compiling, and then stops, throwing the exception shown in [Figure A-2](#).

Figure A-2 On JACE-2 series platform, missing “maxHeap” license feature causes exception



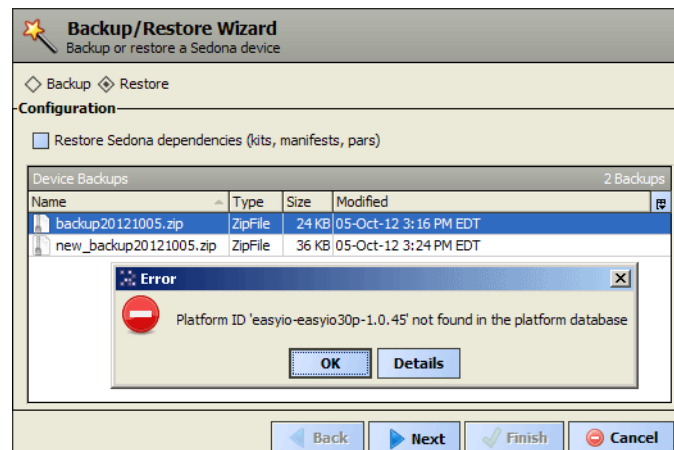
In order for the JACE-2 series platform to be able to compile scode it requires the maxHeap license feature. If you have a JACE that does not have this feature, you can contact your product vendor to find out if the license can be upgraded.

Provisioning fails when Sedona Environment not configured on platform

This can occur when you attempt to use Sedona Tools under a networked device on a JACE when the platform has not been configured with a “Sedona Environment”. The Put (Application Manager), Restore (Backup/Restore Tool), and Kit Manager operations do not work.

The error messages that display when you access the Sedona Tools indicate that the referenced platform manifest or “platform ID” for the connected Sedona device is missing from the platform database, similar to the one shown in [Figure A-3](#).

Figure A-3 Error message when no Sedona Environment configured on JACE



You can correct the problem by connecting to the JACE platform and using the Sedona Environment Manager tool to install the environment files (kits/manifests/PARs) that you need for the Sedona devices on your system. For more information, see the “Sedona environment Overview” section, *NiagaraAX Sedona Framework Networks Guide*.

Components and Plugins

Sedona Tools components

The Sedona Tools is the container for the three Sedona provisioning tools. Located under each SedonaDevice, the component is present in any new SedonaDevice you create. If the component has been removed, you can copy the Sedona Tools component from the **sedonanet** palette into your networked station's SedonaDevice. The default view of the Sedona Tools is the Sedona Tools View.

Sedona Tools plugins (views)

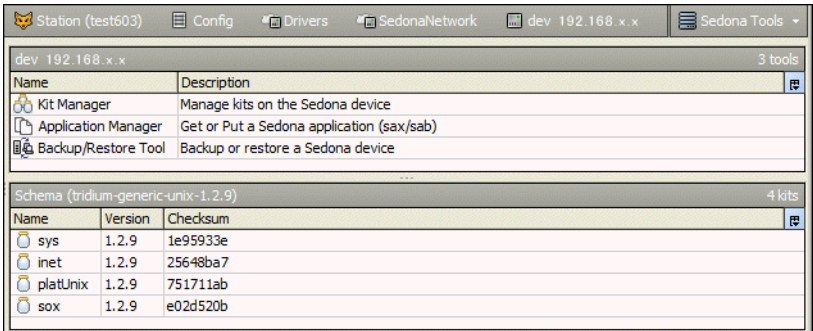
The following views are found in the Sedona Tools:

- [Sedona Tools view](#)
- [Backup/Restore Tool view](#)
- [Application Manager view](#)
- [Kit Manager view](#)

Sedona Tools view

The Sedona Tools nav container view is the default view of the Sedona Tools—double-click the Sedona Tools node to access the three provisioning tools, or select you can select the individual provisioning tools from the Views menu.

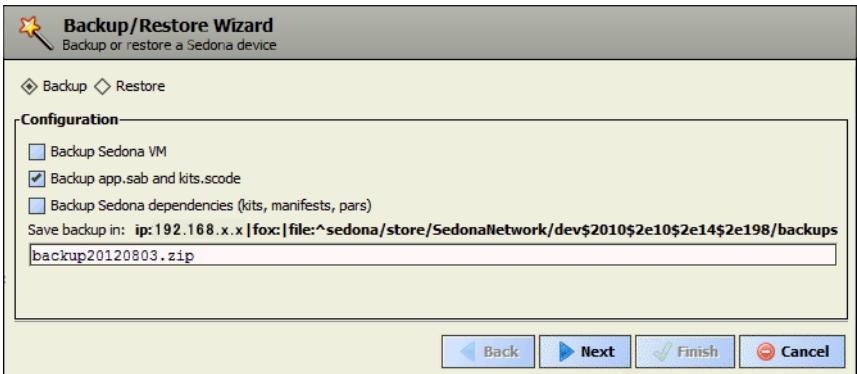
Figure B-1 Sedona Tools view



Backup/Restore Tool view

Creates backup files for Sedona Framework device components as zip files and restores the backups. These backup files can contain the Sedona virtual machine (SVM), the app.sab and kits.scode, as well as the kits, manifests, and PAR files. When creating a backup of a device, the data is saved in a zip file in the file system hierarchy under the device name. When restoring a backup to a device, the contents of the selected zip file are saved to the connected device.

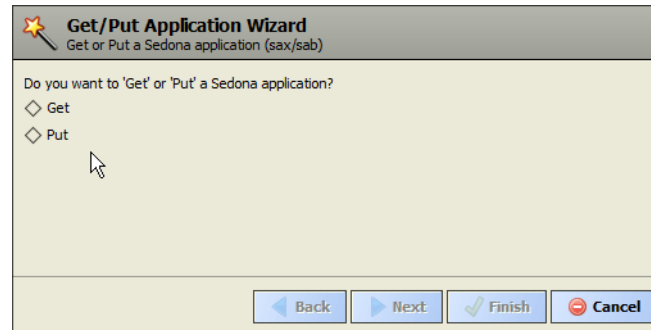
Figure B-2 The Backup/Restore Tool dialog box (provisioning through networked station)



Application Manager view

Manages applications on a Sedona Framework device. Application Manager tool is used to copy an application *from* a Sedona Device using the Get operation or *to* a Sedona Device using the Put operation. When launched, Application Manager displays the Get/Put Application wizard, shown below, which leads you through the steps of each procedure.

Figure B-3 The Application Manager tool



Kit Manager view

Manages the kits.scode on a Sedona Framework device. The Kit Manager tool allows you to add and remove kits, as well as to change version levels of kits deployed in the scode running on the device. Also, you have the option to force the kits.scode to be rebuilt. When launched, the Kit Manager analyzes the kits installed on a device. A progress bar displays in the view during this analysis. Afterwards, the Kit Manager view displays featuring the Kits table and Verifier table, as shown in here.

Figure B-4 Kit Manager view

