CA Release Automation - Integrations

Rapid Development Kit

Date: 11-Feb-2018

Table of Contents

Rapid Development Kit

Rapid Development Kit 1.0.6

The Rapid Development Kit (RDK) is a complementary tool to CA Release Automation. The RDK enables non-JAVA programmers to create Action Packs to provide quick, consistent code generation and packaging to customize environments. N ew actions and action packs are released on a monthly basis.

Use of this tool assumes that end users:

- Have an intermediate knowledge of CA Release Automation
- Are an SME on the specific application or system that is orchestrated through CA Release Automation
- Are skilled in using CLI/Scripting, such as for cmd, bash, and PowerShell, and parsing output data through RegEx
- Are skilled in the use of RESTful Web Services and parsing output data through JSONPath or XPath

Features

- RDK supports Command Line, Scripts, and RESTful APIs.
- RDK generates ready-to-use Action Packs with support for Windows and Linux.
- RDK generates Action Pack documentation templates.

What's New

The following updates were made for RDK 1.0.6

The inline Help section has been fixed, reflects the correct version number.

The following updates were made for RDK 1.0.5

- The "validation-api" jar file error no longer occurs during installation.
- CLI actions now accepts the use of only one input field.

The following updates were made for RDK 1.0.4

- Added a Prefix option to ensure that, when query parameters values are NULL, the respective input parameters are ignored after execution. The input parameter prefixes will not show in the URI or command line.

The following updates were made for RDK 1.0.3:

- The Settings function now includes an option to publish an RDK-generated action pack directly to an RA server.
- Clone functionality moved to the Add Action Pack panel.

The following updates were made for RDK 1.0.2:

- Added instructions on how to upgrade the RDK version to 1.0.2.
- The Setting function now provides two menu items, Local folders and GIT. Read-only fields moved to About Info.
- GIT-related functions button added to the Action Pack details screens.

The following updates were made for RDK 1.0.1:

- Added an Array option for CLI, REST, and Script action types.
  - Added a Delimiter option for Arrays.
- Fields now update in a loop.
- Removed RDKCommon from action jar files. RDKCommon.jar is now an independent dependency.
- Output parameters can now be placed into error conditions.

RDK Workflow

The RDK interface has simple navigation and input to create and maintain action packs and their respective actions. The RDK basic workflow is as follows:

- Install and Configure the RDK (see page 8)
- Create or Modify Action Packs in RDK (see page 12)
- Create or Modify Actions in RDK (see page 13) and maintain their properties:
  - Define input parameters
  - Define the execution payload that you want to run and map the input parameters as arguments to the execution payload.
  - (Optional) Define extra output.

- (Optional) Define extra execution results.

- Export and Run RDK Action Packs (see page 19):
  - Publish by exporting as a jar file.
  - (Optional) Export a documentation .xml file.
  - Publish directly to an RA server

RDK Training Video

The following training video compliments the RDK Workflow articles by providing a visual reference to guide you along.

Install and Configure the RDK

Install and configure the CA Release Automation RDK interface to create new action packs or modify the properties of an existing action packs.

Requirements

Verify that your system meets the minimum requirements to install the CA Release Automation Rapid Development Kit (RDK).

System requirements:

- Windows 2008 R2 or later
- RHEL 6.4 or later
- CA Release Automation 4.7.1 or later
- Oracle JDK 7 64-bit
- JAVA_HOME variable set to JDK installation folder
- Apache Tomcat 7 64-bit. Do not use the Tomcat instance included with CA Release Automation.

Prepare to Install the Rapid Development Kit

Before you install the CA Release Automation RDK:

- Verify the system requirements.

- Download the RDKCommon-<version>.jar file from the CA Support (https://support.ca.com/) site. This file allows you to run action packs generated by RDK in Release Automation. Import the RDKCommon-<version>.jar as a dependency into Release Automation.
- Download the following CA Release Automation RDK installation files from the CA Support (https://support.ca.com/) site.
  - Windows:
    - RDK 1.0.<version> Windows Installer
    - RDK 1.0.<version> Windows Third-Party Installer
  - Linux
    - RDK 1.0.<version> Linux Installer
    - RDK 1.0.<version> Linux Third-Party Installer

> **Note**: Do not install the RDK on a production system where Release Automation is installed. Only install the RDK on a development or test system.

Install the Rapid Development Kit

Installation is a two-part installer where the main installer launches the third-party component installer.

Follow these steps:

1. Run for Windows or Linux, respectively:
   - ca_ra_rdk_windows-x64_<version>.exe
   - ca_ra_rdk_windows-x64_<version>_JRE1_7.exe Includes the required built-in Java 7.

     For Linux users: installation requires root privileges.
   - ca_ra_rdk_unix_<version>.sh
   - ca_ra_rdk_unix_<version>_JRE1_7.sh Includes the required built-in Java 7

   The installer for the third-party components runs automatically.
2. When the installer prompts you, specify the Tomcat 7 installation folder.

3. When the installation is complete, restart the Tomcat server.
   Note: If Tomcat is install as a non-root user, the following manual steps are required: Change the owner of the following folders under <Tomcat Home>/webapps; /rdk and /RDKServer
   Example: chown -R <Tomcat Owner>:<Tomcat Group> <Tomcat Home>/webapps/RDKServer

Upgrade the Rapid Development Kit

The RDK supports direct upgrades from previous releases. Upgrades occur automatically when you run the installer on a system where the previous release is installed.

Follow these steps:

1. Download the installer from the CA Support (https://support.ca.com/) site.
2. Run the installer for the RDK to upgrade the installation.
3. Retrieve the file from the RDK installation at: <tomcat_dir>\webapps\RDKServer\WEB-INF\lib. Alternatively, download the RDKCommon-<version>.jar file from the CA Support (https://support.ca.com/) site.
4. Import RDKCommon-<version>.jar as a dependency into Release Automation.

If you have an imported action pack that was generated previously from the RDK, perform the following steps after you upgrade the RDK:

1. Identify the action packs that were generated from the RDK.
2. Regenerate the action packs with the newest version of the RDK. Delete action packs that could not be regenerated from CA Release Automation
3. Re-import the action packs into CA Release Automation.
4. Ensure that there are no older versions of the action packs in CA Release Automation.

Launch the Rapid Development Kit

To launch the RDK, perform one of the following steps:

- Access the hostname of the tool, such as http://<host name>:8080/rdk.

  Note: Do not use a localhost or FQDN for hostname to access RDK.

- Select the desktop shortcut. By default, a shortcut is created during installation.

If you are not logged in when you launch the RDK, you are prompted to log in. The installation default user name and password are superuser and suser, respectively. Optionally, to change the user name and password, modify the **tomcat-users.xml** file in the conf folder in the Tomcat installation folder.

Settings

The Settings function allows you to reference or modify certain directory settings for the RDK on your system. To launch the Settings, click the gear icon in the upper right of the UI.

Local Folders

The Set Local Folders function allows you to reference or modify certain directory settings for the RDK on your system. To launch Set Local Folder, click Settings and select Local Folder. You can modify and save the following fields:

- The Execution Directory where the CLI commands are locally executed.
- The named Base Package with applicable action packs.

RA Servers

The RA Servers option allows you to configure your servers to publish completed action packs directly to CA Release Automation.

Follow these steps:

1. Click Settings and select the RA Servers option.
2. From the SET RA Servers panel, add the server address, a user name, and a password and click Save.
   The new server displays on the left. To modify an RA server, select the server from the left panel, modify the field data, and click Save.
3. (Optional) To test the server connection, display the server and click Test Connection.
   A pop-up window indicates success or failure.

About

Click the information icon in the upper right of the UI to view the About option. The information includes:

- The RDK version information
- A Working Directory that stores your action packs and files, including distribution, libraries, and source.

- The JAVA Home location

Create or Modify Action Packs in RDK

You can use the CA Release Automation RDK interface to new action packs manually or modify the properties of an existing action packs. You can also duplicate exiting action packs and used the duplicates as a template for a new or deleted action pack.

Create Action Packs in RDK

- To create an action pack and assign properties, click +Add Action Pack, complete the screen fields, and click Save.
- To copy an action pack and the assigned actions, select the action pack and click the Duplicate option. Name the duplicate action pack and click Duplicate. This function lets you make subsequent versions of an action pack and actions, and preserve the original version/template.
- To clone an existing action pack repository on the GIT server to a local location, click +Add Action Pack. Select CLONE From Git, enter the Git Repository, and click Save.

Note: To use the Clone function, first create an empty action pack in the RDK with an identical name to the GIT repository. Spaces between words and uppercase and lowercase letters are acceptable when you name the action pack.

Properties

The properties that are designated for action packs in the RDK correspond to the properties used in the CA Release Automation interface. These properties include its category, description, and version. CA Release Automation uses the Primary Category when browsing for actions.

If the Primary Category for the action pack is blank, the Action Category that is assigned to an action is used. This allows one action pack to contain actions that are split across multiple categories. You can also the use .(dot) annotation to create nested categories. If a Primary Category is specified at the action pack level, this category becomes the parent category for all actions within that action pack.

Modify Action Packs in RDK

- To open the properties of an existing action pack, click the action pack icon.
- To delete an action pack, select the action pack and click the red X above the action pack icon.

- (Optional) Update the cube icon, for example with an applicable vendor icon. To select an alternative .jpeg file for the display, click Browse.

GIT-related functionality

The action pack modification interface also includes GIT-related functionality. The GIT button provides a drop-down list to create repositories and update to repositories.

- **Create**. This option creates a repository on the GIT server. This option pushes the local files of the current action pack to the repository.
- **Update**. This option pushes local changes to the GIT repository.

Create or Modify Actions in RDK

Actions are the building blocks that are grouped in a defined sequence within a flow. You add actions or modify existing actions within an action pack. The RDK lets you assign a name, a description, and an Action Category to actions. The Action Category is a subcategory of any Primary Category that is defined at the Action Pack level. Typically, when you integrate with a vendor product, the product only supports one type, such as CLI. In these cases, all the actions that are defined in the action pack are CLI actions. Alternatively, different action types can exist within a single action pack.

Create each action:

Once an action is created in the RDK, the options below each action let you delete the action, or move and copy the action between action packs.

To open the Action Details pane, click Add Action and select CLI, SCRIPT, or RESTful. You can also choose an existing action to modify.

- **Name** - The name is displayed in the CA Release Automation Actions browser.
- **Description** - The description is displayed in the CA Release Automation Actions browser.
- **Action Category** - Groups actions into subcategories by type or class. By default, an action is associated to the **Primary Category**, as defined for the parent action pack.

Define Input Parameters

Each action is defined with one or more Input Parameters. These parameters are flexible. For example, you can use an entire URL as an input parameter. The parameters include:

- Name
- Type (String, Password, Boolean, Integer, Float, or Selection)
- Required flag
- Array, add items to the array
- Description
- Prefix

  Note: If an input parameter is for a CLI or RESTful Action, add the prefix to this field. If the parameter value returns as NULL and if this field has the prefix added, the prefix will not display after execution.

  Example: A new input parameter is defined with -version=${value} and dragged into a RESTful Action. If the value comes back NULL, the prefix displays in the URI and command line while the value does not display. Add -verson to the Prefix field to ensure that the NULL value parameter is ignored.
  For a RESTful action, this option functions for the Request URI field. This option does not function for Request Headers or Body Content / include the Raw body.

- Default Value

To create an additional Input Parameter, click the plus (+) button.

Note: The order in which input parameters are added corresponds to the parameter appearance in the RA action properties. To change the order, Drag and drop the parameters.

Define CLIs or Scripts to Run

After you define Input Parameters, you determine what CLI or script to run. You can create Windows and Linux CLI/Scripts and can upload your own pre-made script. While the Input and Output parameters stay relatively static across the Action Types, the scripting (the content of the action) varies. Creating action packs through the RDK does not require JAVA or programming knowledge. The tool assumes a user has basic scripting knowledge.

CLI

You have two options for scripting: manual input or script upload. The scripting pane elements are:

- **Execution Path**
  Defines the directory from which the script is executed.

- **Available Input parameters**
  Includes the parameters that are defined through the Input Parameters section, such as appName, location, and password. Drag and drop a defined Input Parameter into the string in the Command field. Placed Input Parameters can be arranged in a specific order. If no value is entered, a default value is provided.

- **Command**
  The script to run for the action pack.
  Note: By default, RDK executes the commands or script for which an interpreter is present. For example, if you run Python, you use the CLI Action Type. Next, explicitly call the Python interpreter and provide a command or script file reference and arguments as input. For example, python -c "print ('hello')".

- **Upload script**
  Allows you to upload a script or some other scripting language. Once uploaded, the script is referenced as ${SCRIPT_NAME} in the Command option.

- **Execute**
  Allows you to run the script locally and validate the script output for accuracy.

After you define Windows script commands, you can similarly define for Linux script commands.

Advanced options

- **Wait for process to finish**
  By default, there are no time constraints on to run script process. If the system times out during the process, a duration time (in seconds) can be set here. Clear the check box and set the **Timeout duration**. If the time-out lasts longer than the set time, the script exits and is marked as a failure.

- **Overwrite output file**

- **Environment variables**
  (Name = Value)

SCRIPT

There are fundamental differences between the CLI and SCRIPT action types.

- **CLI Action Type**
  Executes both commands and scripts, and assumes that the script accepts command-line arguments.

- **SCRIPT Action Type**
  Executes a script directly against the OS default interpreter (cmd.exe or bash) or PowerShell (Windows only) and enables inline variable substitution.

  

  Note: For SCRIPT action types, the script pane elements vary depending on your selection of Default OS or PowerShell. Other elements, such as Available Input Parameters, remain the same.

Default OS

- **Execution Path**
  Defines the directory from which the script is executed.

- **Available Input parameters**
  Drag and drop a defined Input Parameter into the script field. Placed Input Parameters can be arranged in a specific order. If no value is entered, a default value is provided.

- **Script**
  Allows he script to run for the action pack.

- **Upload script**:
  Allows you to upload a script or some other scripting language.

- **Execute**
  Allows you to run the script locally and validate script output for accuracy. By default, RDK executes the commands or script for which an interpreter is present.

After you define Windows script commands, you can similarly define for Linux script commands.

- **Advanced options**

  - **Wait for process to finish**.
    By default, there are no time constraints on a running script process. If the system times out during the process, a duration time (in seconds) can be set here. Clear the check box and set the **Timeout duration**. If the time-out lasts longer than the set time, the script exits and is marked as a failure.

  - **Overwrite output file**

  - **Environment variables**
    (Name = Value)

PowerShell

- **Working Directory**
  The directory from which the script is executed.

- **Advanced options**

- **Execution Timeout**
  (in seconds)

- **PowerShell Options**
  (Name = Value)
  Example: ExecutionPolicy=Bypass

RESTful

Elements in the RESTcall section, such as Available Input Parameters, are the same across the action types. Others are unique to the RESTful type.

- **Available Input parameters**
  Drag and drop a defined Input Parameter into the Request URI field.

- **Execute RESTful Call**:
  Allows you to execute locally and validate its output for accuracy. When executed, the RDK executes the given request method on the Request URI.

Target

- **Request URI**
  The Uniform Resource Identifier with which to communicate.

- **Request Method**

- GET, POST, PUT, or DELETE.

- **Request Timeout**

- The time limit for the request, in seconds.

- **Request Headers**

- **Body Content / Content Headers**
  This data does not display with GET requests.
  Under Request Parameters, guided works with key value pairs, such as myKey=myValue. When sending structured data, such as JSON or XML, use RAW instead:

  1. Under Body Content, click the Switch to RAW button.
     The Advanced Rest Client screen displays.

  2. Under the RAW tab, enter complex structured data, such as JSON or XML.
     The JSON data displays in the RAW Body field.

Authorization

- You must choose an Authorization header from the following:

  - No Authentication

- Basic Authentication
- NTLM
- Digest Authentication.

Note: If no authentication is chosen and you attempt to save the action, the system flags the action as incomplete. Basic Authentication automatically adds a user and password to the Input Parameters listing.

Define Output Parameters and Filtering

Each action is defined with one or more Output Parameters.

Note: If nothing is defined, the default output parameter is Execution Results and the action produces all output.

Once defined, the subsequent action inputs use the output values. To define the output, assign a value to the output. This output can be validated and displayed in the Script Execution Results. The parameters are:

- Output Parameter Name
- Type (for example, String)
- Array
- Description
- Filter Type (XPATH, JSONPath, Delimiter, and Reg Exp) (Note: N/A if the source is Exit Code)
  Note: You can create an array using any Filter (action) Type. For example, you can generate a REST type array using the JSONPath action type.

Note:
- The Delimiter option only applies to array creation.
- When using the XPath, JSONPath, and Delimiter options, ensure that queries retrieve data that is homogeneous. The action fails if, for example, you attempt to convert a string into an integer.

- Source (Standard Out, Standard Error, Exit Code)
- Find Using Regular Expression

- (Optional) Create input parameter for output file location

To create an additional Output Parameter, click the plus (+) button.

Define Execution Results and Error Messages

The Results and Error Conditions section gives you control over how to define and display evaluation results. This function is useful when you run, for example, a successful API Call that is tagged as failed based on the output.

You define an error condition that is based on the action results, such as determining under which conditions an action is marked as a failure. The defined output parameters are combined with an Operator and Value, such as:

Execution Output (parameter) -- **Contains** (operator) -- **Exception** (value)

After you execute a script and the script returns, it shows success or failure that is based on the error conditions. If none of the error conditions match, then it returns the success message. If the conditions are not defined, the system defaults to successful.

Export and Run RDK Action Packs

To export an action pack, use the Export function on the action pack interface. To run the action pack, import the action pack through Agent Management in CA Release Automation. Use the following functions to export, run and publish action packs:

- **Export as Jar**

Generates code and prompts you to download a .jar file locally. You do not need JAVA code or source generation knowledge to compile the .jar file. To import the action pack directly into the environment, export an action pack as a .jar file. Through Action Management in CA Release Automation, you can browse, open, and import the .jar file.

- **Export Documentation**

Generates code and prompts you to download the documentation XML file. The file contains CA Release Automation action pack properties, and details for each action. These details include name, description, input parameters, and description.

- **Run the Action Pack**
  Enables you to run an action pack that is generated by RDK in CA Release Automation. Download the RDKCommon-<version>.jar file from the CA Support (https://support.ca.com/) site and import the file through Agent Management in CA Release Automation.

- **Publish to RA Server**

Enables you to publish your action pack directly to the RA server.

Note: Ensure that your RA servers are configured to use this option. For more information, see Install and Configure the RDK (see page 8).

Follow these steps:

1. Select the action pack, click the Export drop-down list, and select Publish to RA Server.
2. From the RA Servers drop-down list, select a server to which to publish.
3. Verify that the action pack publishes through the inventory listing under Action Management on the Administration tab.

RDK - Known Issues

The following items are known issues with the RDK:

- The screen does not scroll when you drag input parameters beyond the visible area. You can manually enter the variable name of input and output parameters into supported fields such as ${variableName} and they are then parsed.
- Once you name an action, the action name cannot be changed. To change an action name name, make a copy of the action, rename the action, and delete the original action.
- CA Release Automation does not support default values and marking values as required:
  - Do not enter a default value if a value is marked as required.
  - Do not mark a value as required if it has a default value.

RDK - Acknowledgements

The Rapid Development Kit (RDK) uses the following third-party software under license:

Apache Commons Codec

This product includes Apache Commons Codec 1.3 and is distributed in accordance with the Apache Software License v.2.0.

Apache HttpComponents Core

Copyright 2005-2011 The Apache Software Foundation

Apache Commons httpmime

This product includes Apache Commons httpmime 4.1

Apache HttpComponents HttpMime

Copyright 1999-2011 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This project contains annotations derived from JCIP-ANNOTATIONS

Copyright (c) 2005 Brian Goetz and Tim Peierls. See http://www.jcip.net

Apache Mime4J

This product includes Apache Mime4J 0.6, which is distributed in accordance with the following license agreement:

Apache JAMES Mime4j

Copyright 2004-2009 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Software Foundation

Parts of this product include software developed by the Apache Software Foundation. The Apache software is distributed in accordance with the following license agreement:

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity

authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution.

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The

contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability.

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Apache wss4j

This product includes Apache wss4j 1.5.8

Apache WebServices - WSS4J

Copyright 2004-2009 The Apache Software Foundation

This product includes software developed at The Apache Software Foundation (http://www.apache.org/).

This product includes software Copyright University of Southampton IT Innovation Centre, 2006 (http://www.it-innovation.soton.ac.uk).

Apache XMLSchema

This product includes Apache XMLSchema 1.4.2

This product includes software developed by The Apache Software Foundation (http://www.apache.org/). Portions Copyright 2006 International Business Machines Corp.

Apache xmlsec

This product includes Apache xmlsec 1.4.4

This product contains software developed by The Apache Software Foundation (http://www.apache.org/). It was originally based on software copyright (c) 2001, Institute for Data Communications Systems, <http://www.nue.et-inf.uni-siegen.de/>.

The development of this software was partly funded by the European commission in the <WebSig> project in the ISIS Programme.

CA Inc

CA, Inc. ("CA")

End User License Agreement (the "Agreement") for the CA software product that is being installed as well as the associated documentation and any SDK, as defined below, included within the product ("the Product").

Carefully read the following terms and conditions regarding your use of the Product before installing and using the Product. Throughout this Agreement, you will be referred to as "You" or "Licensee."

You are:

(I) Representing that you are not a minor, and have full legal capacity and have the authority to bind yourself and your employer, as applicable, to the terms of this Agreement;

(II) Consenting on behalf of yourself and/or as an authorized representative of your employer, as applicable, to be bound by this Agreement.

Castor

This product includes Castor software distributed in accordance with the following license agreement:

Copyright 2004-2005 Werner Guttmann

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

google-gson

This product includes google-gson 1.7.1, which is distributed in accordance with the Apache Software License v.2.0.

httpclient

This product includes httpclient 4.1.2, which is distributed in accordance with the following license agreement:

Apache HttpComponents HttpClient

Copyright 1999-2011 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This project contains annotations derived from JCIP-ANNOTATIONS

Copyright (c) 2005 Brian Goetz and Tim Peierls. See http://www.jcip.net

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

HttpComponents HttpCore

This product includes Apache HttpComponents HttpCore 4.1, which is distributed in accordance with the following license agreement:

Apache HttpComponents HttpCore

Copyright 2005-2010 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Install4J

Install4J runtime libraries under license from ej-technologies (http://www.ej-technologies.com).