



# Coverity Platform 2020.12 Web Services API Reference

Copyright 2020 Synopsys, Inc. All rights reserved worldwide.

---

## Table of Contents

1. Coverity Platform Web Services API Reference (v9) .....	1
1.1. Description .....	1
1.2. Sources .....	1
1.3. Overview .....	2
1.4. Audience .....	2
2. Getting Started .....	3
3. Configuration Service Web Service .....	4
3.1. Name .....	4
3.2. Description .....	4
3.3. Type .....	4
3.4. See Also .....	4
3.5. Operations: Configuration Service .....	4
3.6. Complex Types .....	74
3.7. Simple Types: Configuration Service .....	123
4. Defect Service Web Service .....	125
4.1. Name .....	125
4.2. Description .....	125
4.3. Type .....	125
4.4. See Also .....	125
4.5. Operations: Defect Service .....	125
4.6. Complex Types .....	147
4.7. Types and Constants .....	179
5. View Management API .....	187
5.1. view/v1/username/{source_user}/ldap/{source_ldap} .....	187
5.2. view/v1/view/{view_id} .....	189
5.3. view/v1/views .....	191
5.4. views/v1 .....	194
5.5. viewContents/{type}/v1/{view_id} .....	197
6. Error Codes .....	202
6.1. 1000-1099: General .....	202
6.2. 1100-1199: User .....	203
6.3. 1200-1299: Attribute .....	203
6.4. 1300-1399: Project, stream, snapshot .....	204
6.5. 1400-1499: Defect .....	205
6.6. 1500-1599: Internal .....	205
6.7. 1600-1699: Component .....	205
6.8. 1700-1799: Role .....	206
6.9. 1800-1899: File and function .....	206
6.10. 1900-1999: License .....	206
6.11. 2000-2099: Triage store .....	206
6.12. 2100-2199: LDAP .....	207
6.13. 2200-2299: View Notification .....	207
7. Glossary .....	208
7.1. attribute .....	208
7.2. checker .....	208
7.3. CID .....	208

7.4. component .....	209
7.5. defect .....	209
7.6. external reference .....	209
7.7. fix target .....	209
7.8. permission .....	209
7.9. project .....	209
7.10. role .....	210
7.11. snapshot .....	210
7.12. source version .....	210
7.13. stream .....	210
7.14. subcategory .....	210
7.15. target .....	211
7.16. triage .....	211
7.17. triage store .....	211
7.18. version .....	211
8. API History .....	212
8.1. Change Logs .....	212
8.2. Configuration Service Log .....	213
8.3. Defect Service Log .....	221
A. Legal Notice .....	228
A.1. Legal Notice .....	228

---

# Chapter 1. Coverity Platform Web Services API Reference (v9)

## Table of Contents

1.1. Description .....	1
1.2. Sources .....	1
1.3. Overview .....	2
1.4. Audience .....	2

### 1.1. Description

The Coverity Platform Web Services API allows you write Web applications that communicate with Coverity Connect (part of the Coverity Platform installation). The API uses SOAP (Simple Object Access Protocol) to send requests to the server and receive responses to those requests.

Note that there are also several operations which use REST (Representational State Transfer) to receive information about Coverity Connect views. See View Management API for details.

The REST operations are simpler to call, and require less processing. If you can set up a view in the Coverity Connect user interface, you should probably use the View Management API.

This API reference is for the latest version of Coverity Connect. For details, see Chapter 8, *API History*.

### 1.2. Sources

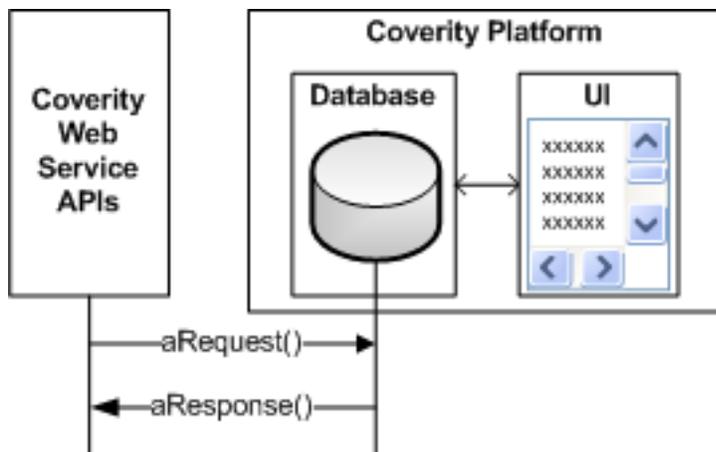
Name	Description
Overview	Introduction to the API, including skills needed by the audience and how to get started with the API.
Configuration Service	API for performing project, stream, user and group (both LDAP and local), role, component/component map, project, attribute, triage store, and other configurations (many of which are also possible through the UI and/or through the cov-manage-im command). It also supports notifications and expiration of unused streams.
Defect Service	API for operations that pertain to CIDs and instances (occurrences) of software issues.
Types and Constants	Enumerations and string-based values that are valid for specified fields and parameters.
Error Codes	Descriptions of error codes returned for invalid Web service requests.
Glossary	Definitions of terms used in this API.
REST Web Services	API for view management operations.
Chapter 8, <i>API History</i>	Lists of changes to the API since the preceding version. The current version is listed first.

Name	Description
Legal Notice	Coverity legal notice.

### 1.3. Overview

The Coverity Web Services API allows you write Web applications or scripts that communicate with Coverity Connect database.

The API uses SOAP (Simple Object Access Protocol) to provide access to some of the functionality of the server. Access includes the creation, deletion, retrieval, and modification of various objects (such as users, projects, streams, and so on) in the database. SOAP uses XML to format and transport messages to and from the server.



You need to create applications or scripts that support API requests and handle responses from the database. See Getting Started for related information.

### 1.4. Audience

This reference assumes that you are familiar with the key concepts of Coverity Platform and that you have the following background:

- Knowledge of Web Services and WSDL files
- An understanding of SOAP
- An understanding of XML
- Knowledge of a language that supports SOAP bindings

Examples: Perl, Python, C/C++, Java, C#.

For more on Coverity Platform, see the *Coverity Platform Use and Administration Guide*.

---

## Chapter 2. Getting Started

### To get started with the API:

1. Use the free **SoapUI** software (<http://www.soapui.org> or <http://www.smartbear.com>) to download the WSDL files and create test suites with query templates for the calls. You can also use this software for checking the return XML results. The URLs for the WSDL files are as follows:

- Configuration service:

`http://<cov_connect_server>.<domain>:<port>/ws/v9/configurationservice?wsdl`

(for example, `http://my.cov_connect.domain:8080/ws/v9/configurationservice?wsdl`)

- Defect service:

`http://<cov_connect_server>.<domain>:<port>/ws/v9/defectservice?wsdl`

(for example: `http://my.cov_connect.domain:8080/ws/v9/defectservice?wsdl`)

2. Write your own scripts in the scripting language of your choice, and use the appropriate libraries or modules for handling the data objects and calls. The scripts need to use a SOAP library (for example, SOAP::Lite with Perl or suds with Python). Working examples with source code are provided in the example directory, in Python, Perl, and Java. Each zip file contains a Readme file with information on how to run (and build, if necessary) these examples. **Note:** Coverity Connect uses WSSE for authentication, not simple HTTP authentication.

---

## Chapter 3. Configuration Service Web Service

### Table of Contents

3.1. Name .....	4
3.2. Description .....	4
3.3. Type .....	4
3.4. See Also .....	4
3.5. Operations: Configuration Service .....	4
3.6. Complex Types .....	74
3.7. Simple Types: Configuration Service .....	123

### 3.1. Name

Configuration Service

### 3.2. Description

API for performing project, stream, user and group (both LDAP and local), role, component/component map, project, attribute, triage store, and other configurations (many of which are also possible through the UI and/or through the cov-manage-im command). It also supports notifications and expiration of unused streams.

### 3.3. Type

SOAP

### 3.4. See Also

Operations , Complex Types , Simple Types

### 3.5. Operations: Configuration Service

#### 3.5.1. Description

Operations in the Configuration service.

#### 3.5.2. Operations

Name	Description
copyStream	Make a copy of a stream. Does not copy stream role assignments.
createAttribute	Create an attribute.
createComponentMap	Create a component map with or without one or more components.
createGroup	Create a user group.
createLdapConfiguration	Create an LDAP configuration for users and groups.
createProject	Create a project.
createRole	Create a role.

Name	Description
createStream	Create a stream that is not associated with any project. The new stream will appear in the UI as one of the <i>Other Streams</i> .
createStreamInProject	Create a stream in a specified project.
createTriageStore	Create a triage store.
createUser	Create a user.
deleteAttribute	Delete an attribute.
deleteComponentMap	Delete a component map.
deleteGroup	Delete a user group.
deleteLdapConfiguration	Delete an LDAP configuration.
deleteProject	Delete a project.
deleteRole	Delete a role.
deleteSnapshot	Delete a snapshot.
deleteStream	Delete a stream.
deleteTriageStore	Delete a triage store to which no streams are associated.
deleteUser	Delete a user.
executeNotification	Initiates a view notification to preconfigured recipients. The view must belong to the user who calls this operation.
getAllLdapConfigurations	Retrieve all LDAP configurations.
getAllPermissions	Retrieve the complete list of permissions that can be associated with a role.
getAllRoles	Retrieve a list of all roles.
getArchitectureAnalysisConfiguration	Returns a string carrying configuration information for use with the Architecture Analysis product. It is only intended for internal use by that product.
getAttribute	Retrieve the properties of a specified attribute.
getAttributes	Retrieve a list of all attributes.
getBackupConfiguration	Retrieve the schedule for automated backup of the Coverity Connect database.
getCategoryNames	Retrieve all known checker category names, as localizedValueDataObj.
getCheckerNames	Retrieve all known checker names.
getCommitState	Find out whether the database will accept new commits of analysis results.
getComponent	Retrieve the properties of a component.
getComponentMaps	Retrieve a list of component maps that matches a component name pattern.

Name	Description
getDefectStatuses	Retrieve the list of Status attribute values that can be associated with a software issue.
getDeleteSnapshotJobInfo	Find out whether a snapshot deletion process succeeded.
getDeveloperStreamsProjects	Get a list of project specifications in developer streams (for all such projects or for a filtered set of such projects).
getGroup	Retrieve the properties of a user group.
getGroups	Get a list of groups.
getLdapServerDomains	Retrieves the host name or host IP of one or more LDAP servers.
getLicenseConfiguration	Retrieve details on your Coverity Connect license.
getLicenseState	Find out whether Coverity Desktop analysis is enabled through your license.
getLoggingConfiguration	Retrieve information about your current Coverity Connect configurations for logging.
getMessageOfTheDay	Get the message of the day.
getOutputFileForSnapshot	Get a specified output file for a specified snapshot.
getProjects	Get a list of project specifications (for all projects or for a filtered set of projects).
getRole	Retrieve the properties of a role, including its associated permissions.
getServerTime	Retrieves the current date and time from the server.
getSignInConfiguration	Retrieve sign-in settings. These configurations are identical to Sign In Settings (a set of System Configuration settings) in the Coverity Connect UI.
getSkeletonizationConfiguration	Retrieve the configuration for the process that purges snapshot details. Purging these details can help you reduce and maintain the database size.
getSnapshotInformation	Retrieve information about a snapshot in a stream.
getSnapshotPurgeDetails	<b>Deprecated in v8:</b> Use getSkeletonizationConfiguration() instead to retrieve the configuration for the process that purges snapshot details. Purging these details can help you reduce and maintain the database size.
getSnapshotsForStream	Retrieve a set of snapshots that belong to a specified stream.
getStandardAttribute	Retrieve the properties of a specified standard attribute.
getStandardAttributes	Retrieve a list of all standard attributes.
getStreams	Retrieve a set of streams.
getSystemConfig	Retrieves system configuration properties, including properties of the database with which this API communicates.

Name	Description
getTriageStores	Retrieve a set of triage store specifications, including stream associations.
getTypeNames	Returns a list of all known defect types, as localizedValueDataObj.
getUser	Retrieve a user by user name.
getUsers	Get users (filtered or unfiltered).
getVersion	Retrieve the version of Coverity Connect.
importLicense	Import your Coverity Connect license file, license.dat.
notify	Send an email notification to a specified user.
refreshLdapGroup	Refresh an LDAP group.
setAcceptingNewCommits	Control whether the database will accept new commits of analysis results.
setArchitectureAnalysisConfiguration	Set the value that getArchitectureAnalysisConfiguration returns.
setBackupConfiguration	Set a schedule for automated backup of the Coverity Connect database. The name of the backup file looks something like the following: CIM.2013-12-04.10-35.backup
setLoggingConfiguration	<p>Enable or disable logging options for Coverity Connect.</p> <p>Coverity Connect automatically logs system event information to the cim.log file. You can increase the amount of information that Coverity Connect records to this file by enabling additional logging options to work with Coverity Support on an issue. Coverity recommends that you leave all of the logging options disabled and only enable them by request from Coverity Support.</p>
setMessageOfTheDay	Set the message of the day.
setSkeletonizationConfiguration	Configure the process that purges snapshot details. Purging these details can help you reduce and maintain the database size.
setSnapshotPurgeDetails	<b>Deprecated in v8:</b> Use setSkeletonizationConfiguration() instead to configure the process that purges snapshot details.
updateAttribute	Update an attribute specification.
updateComponentMap	Update one or more properties of a component map.
updateGroup	Update a group specification.
updateLdapConfiguration	Update an LDAP configuration.
updateProject	Update a project specification.
updateRole	Update a role specification.
updateSignInConfiguration	Update sign-in settings. These configurations are identical to Sign In Settings (a set of System Configuration settings) in the Coverity Connect UI.
updateSnapshotInfo	Update a snapshot.

Name	Description
updateStream	Update a stream specification.
updateTriageStore	Update a triage store specification.
updateUser	Update a user specification.

### 3.5.3. Operation: copyStream

#### 3.5.3.1. Name

copyStream

#### 3.5.3.2. Description

Make a copy of a stream. Does not copy stream role assignments.

#### 3.5.3.3. Parameters

projectId

**Type:** projectIdDataObj

Identifier for a project.

Field name	Type	Description
name	string	Required. Name of the project.

sourceStreamId

**Type:** streamIdDataObj

Identifier for a stream.

Field name	Type	Description
name	string	Required. Name of the stream.

#### 3.5.3.4. Output (Literal)

The output of this operation is the argument copyStreamResponse having the structure defined by the following table.

Name	Type
return	streamDataObj

#### 3.5.3.5. Remarks

To assign roles or change the automatically generated stream name, stream associations, and/or other stream properties, see updateStream().

### 3.5.4. Operation: createAttribute

#### 3.5.4.1. Name

createAttribute

#### 3.5.4.2. Description

Create an attribute.

#### 3.5.4.3. Parameters

attributeDefinitionSpec

**Type:** attributeDefinitionSpecDataObj

Field name	Type	Description
attributeName	string	Name for the attribute. Required when using createAttribute().
attributeType	string	The type of attribute. Required when using createAttribute().
attributeValueChangeSpec	attributeValueChangeSpecDataObj	For a LIST_OF_VALUES attribute type only: The set of values available to the attribute.
defaultValue	string	For a LIST_OF_VALUES attribute type only: The default attribute value.
description	string	Description of the attribute.
showInTriage	boolean	If <i>true</i> , makes the attribute available for use in the Triage pane of the UI.

### 3.5.5. Operation: createComponentMap

#### 3.5.5.1. Name

createComponentMap

#### 3.5.5.2. Description

Create a component map with or without one or more components.

#### 3.5.5.3. Parameters

componentMapSpec

**Type:** componentMapSpecDataObj

Field name	Type	Description
componentMapName	string	Required. Name of the component map.

Field name	Type	Description
componentPathRules	componentPathRuleDataObj	Pattern matching a path to set of files to associate with a component for the component map. Multiple component map rules allowed.
components	componentDataObj	Specification of the component that you are associating with the componentmap. Multiple component associations allowed. When <i>updating</i> (not creating) components, you must specify the complete list of your components, including the "Other" component because any component in the component map that does not appear in the update list will be deleted. Alternatively, you can pass <b>null</b> to retain all existing components during an update.
defectRules	componentDefectRuleDataObj	Default owner of the specified component. Each component can have an owner.
description	string	Description of the component map.
forceDeleteComponents	boolean	When set to <code>true</code> , components that do not appear in the update list and are chosen to be included in hierarchies will be deleted and removed from the hierarchies. When set to <code>false</code> , components that do not appear in the update list and are chosen to be included in hierarchies will <i>not</i> be deleted. In the latter case, an error message is returned.

### 3.5.6. Operation: createGroup

#### 3.5.6.1. Name

createGroup

#### 3.5.6.2. Description

Create a user group.

#### 3.5.6.3. Parameters

groupSpec

**Type:** groupSpecDataObj

Field name	Type	Description
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group. Required when using createGroup() for an LDAP group. Maximum of one <i>domain</i> specification allowed.
local	boolean	Defaults to <i>true</i> , specifying a local (non-LDAP) group. If an LDAP group, set to <i>false</i> when using createGroup(). Maximum of one <i>local</i> specification allowed.
name	string	Name for the user group. Required when using createGroup().
roleAssignments	roleAssignmentDataObj	Role to associate with the group at the global level. See getAllRoles(), getRole(), and getAllPermissions(). Zero or more role associations allowed. If updating role assignments, respecify any global type roles that you want to retain.
syncEnabled	boolean	If a local group, set to <i>false</i> when using createRole(). Defaults to <i>true</i> . Applies to an LDAP group only.

### 3.5.7. Operation: createLdapConfiguration

#### 3.5.7.1. Name

createLdapConfiguration

#### 3.5.7.2. Description

Create an LDAP configuration for users and groups.

#### 3.5.7.3. Parameters

ldapConfigurationSpec

**Type:** ldapConfigurationSpecDataObj

Field name	Type	Description
anonymousBind	boolean	Value of <i>true</i> if using an anonymous bind; otherwise, <i>false</i> to indicate a general bind with a shared bind DN.
baseDN	string	LDAP domain name in host format, such as <domain>.com or

Field name	Type	Description
bindName	string	<p>&lt;corp.domain&gt;.&lt;com&gt;, or base DN such as <i>ou=corp</i>.</p> <p>The username needed to access the LDAP server.</p> <p>You can specify the user in <i>username@domain</i> or LDAP DN format.</p>
bindPassword	string	<p>Bind user password for LDAP user queries. Required for authenticated binding requests. If you do not enter a password, and if your LDAP server is configured to accept unauthenticated binding requests, Coverity Connect will attempt to gain unauthenticated access to the LDAP server. If an unauthenticated LDAP connection fails, Coverity Connect will display an LDAP Server Configuration failed message and an explanation of the failure. You can find more information about the failure in the <i>cim.log</i> file.</p>
displayName	string	<p>Name that identifies the LDAP server in a multi-server setup. Appears in the UI.</p>
groupFilter	string	<p>Optional filter to use when importing LDAP groups. Takes a valid RFC format. Example: <i>cn=eng*</i></p>
groupFullName	boolean	<p>If <i>false</i>, group members are stored by <i>userName</i>. If <i>true</i>, members are stored by their full DN. Defaults to <i>true</i>.</p>
groupMember	string	<p>LDAP attribute that defines the members of a group. Group members can be referred to by their DN or <i>username</i>.</p>
groupName	string	<p>Name attribute for the group. For Example: <i>cn</i></p>
groupObjectClass	string	<p>LDAP <i>objectClass</i> value that identifies user groups. For OpenLDAP, the default is <i>groupofnames</i>. For Active Directory, the default is <i>group</i>. This field defines a component of the LDAP user group search query that Coverity Connect creates.</p>

Field name	Type	Description
groupSearchBase	string	The group search base DN to prepend to the base DN. Used for group searches. The base DN is configured as <i>Domain</i> as part of basic LDAP settings. For example: <i>cn=groups, ou=corp</i>
primary	boolean	Value of <i>true</i> if this configuration is the primary one. Defaults to <i>false</i> .
secureConnection	boolean	Value of <i>true</i> if using a secure connection. SSL or TLS protocol required. For information about LDAP server requirements, secure connections, importing LDAP users and groups, and other details about LDAP, refer to <i>Coverity Connect Usage and Administration Guide</i> .
serverDomain	string	The host name or host IP of the LDAP server. The name must be resolvable from the Coverity Connect host.
serverPort	long	The TCP port number where the LDAP server listens for connections.
tlsEnabled	boolean	Value of <i>true</i> if TLS security protocol is used.
userEmail	string	LDAP attribute that maps to the email address of the user.
userFirstName	string	LDAP attribute that maps to the first name (given name) of the user.
userLastName	string	LDAP attribute that maps to the last name (surname) of the user.
userName	string	Username for testing user search settings. Maps to the username in the database.
userObjectClass	string	LDAP <i>objectClass</i> associated with users on the LDAP server. For OpenLDAP, the default is <i>inetOrgPerson</i> . For Active Directory, the default is <i>user</i> .
userSearchBase	string	Relative DN to prepend to base DN to limit a user search. For example, <i>ou=Users</i> will search in <i>ou=Users, dc=ad, dc=coverity, dc=com</i> .

---

### 3.5.7.4. Remarks

## 3.5.8. Operation: createProject

### 3.5.8.1. Name

createProject

### 3.5.8.2. Description

Create a project.

### 3.5.8.3. Parameters

projectSpec

**Type:** projectSpecDataObj

Field name	Type	Description
streams	streamIdDataObj	Name of an existing stream to associate with the project. When updating a project, you must list all of the streams that you want to retain except for streams with the __internal prefix to their name, such as Dynamic Analysis streams. You might see such internal streams listed when you call getProjects(). Zero or more stream name associations allowed. See the remark for additional details.
streamLinks	streamIdDataObj	Name of a new or existing stream link to associate with the project. (Note that a stream link and the stream to which it links must belong to separate projects.) Zero or more stream link name associations allowed.
description	string	Description of the project.
name	string	Name for the project. Required when using createProject().
roleAssignments	roleAssignmentDataObj	Role to associate with the project. See getAllRoles(), getRole(), and getAllPermissions(). By default, the username of the project creator is assigned the <i>projectOwner</i> role for the new stream. If updating role assignments, respecify any that you

Field name	Type	Description
		want to retain. Zero or more role associations allowed.

### 3.5.8.4. Remarks

To create a stream, see `createStream()` and `createStreamInProject()`.

## 3.5.9. Operation: createRole

### 3.5.9.1. Name

`createRole`

### 3.5.9.2. Description

Create a role.

### 3.5.9.3. Parameters

`roleSpec`

**Type:** `roleSpecDataObj`

Field name	Type	Description
<code>deletable</code>	<code>boolean</code>	If the new role can be deleted, set to <i>true</i> . If not, set to <i>false</i> . Required when using <code>createRole()</code> . Maximum of one specification allowed.
<code>description</code>	<code>string</code>	Description of the role.
<code>editable</code>	<code>boolean</code>	Value of <i>false</i> not allowed. Requires value of <i>true</i> when using <code>createRole()</code> . Maximum of one specification allowed.
<code>name</code>	<code>string</code>	Name of the new role. Required when using <code>createRole()</code> .
<code>permissionDataObjs</code>	<code>permissionDataObj</code>	Name of a permission to associate with the new role. See <code>getAllPermissions()</code> . Zero or more permissions allowed. When updating role permissions, respecify any permissions you want to retain.

### 3.5.9.4. Remarks

Needed only if an existing role or set of roles is not associated with the set of permissions you require. See `getAllRoles()` and `getRole()`.

### 3.5.10. Operation: createStream

#### 3.5.10.1. Name

createStream

#### 3.5.10.2. Description

Create a stream that is not associated with any project. The new stream will appear in the UI as one of the *Other Streams*.

#### 3.5.10.3. Parameters

streamSpec

**Type:** streamSpecDataObj

Specification for a stream.

Field name	Type	Description
allowCommitWithoutPassword	boolean	Boolean that indicates whether Coverity Connect will accept new commits of analysis results (through cov-commit-defects) to the stream without requiring a password. When creating or updating a stream, check your company's security policy before setting to <i>true</i> . If set to <i>true</i> , any user can commit analysis results to the stream.
analysisVersionOverride	string	Specifies the updated Coverity Analysis version that may retrieve data from this stream.
autoDeleteOnExpiry	boolean	Set to <i>true</i> if the stream should be deleted after a period of inactivity. The default period is 28 days. (Note that the <i>stream.expiration.inactivity.days</i> period is configurable through the <i>cim.properties</i> file. The stream will not be deleted unless it contains at least one snapshot.) Defaults to <i>false</i> with createStream() and createStreamInProject().
componentMapId	componentMapIdDataObj	Name of the component map with which to associate the stream. Defaults to the <i>Default Component Map</i> with createStream() and createStreamInProject().
description	string	Description of the stream.

Field name	Type	Description
enableDesktopAnalysis	boolean	If true, this stream is able to accept analysis summary information during commit.
language	string	The programming language of the source code files associated with the new stream. Required with createStream() and createStreamInProject().
name	string	Name of the new stream. Required with createStream() and createStreamInProject().
outdated	boolean	If true, the filter applies to streams that have been designated as <i>outdated</i> and are hidden from the Coverity Connect UI. If false, the filter applies to non-outdated streams.
ownerAssignmentOption	string	Specifies the updated Coverity Desktop plug-in version that may retrieve data from this stream.
pluginVersionOverride	string	
roleAssignments	roleAssignmentDataObj	Role to associate with the new stream. By default, the username of the stream creator is assigned the <i>streamOwner</i> role for the new stream. See getAllRoles(), getRole(), and getAllPermissions(). If updating role assignments, respecify any that you want to retain.
summaryExpirationDays	int	Specifies the number of days analysis summaries remain in this stream before being automatically removed from their relative snapshot. If you do not want summaries to be deleted, set summaryExpirationDays to 0. This will reset the value to null, and analysis summaries will not expire.
trriageStoreId	trriageStoreIdDataObj	Identifier for the triage store to associate with the new stream. Required with createStream()and createStreamInProject(). Maximum of one triage store name specification allowed.

Field name	Type	Description
versionMismatchMessage	string	Provides additional instructions to analysis users that are using an incorrect version of Coverity Analysis or the Coverity Desktop plug-in.

#### 3.5.10.4. Remarks

See also `createStreamInProject()`.

### 3.5.11. Operation: `createStreamInProject`

#### 3.5.11.1. Name

`createStreamInProject`

#### 3.5.11.2. Description

Create a stream in a specified project.

#### 3.5.11.3. Parameters

`projectId`

**Type:** `projectIdDataObj`

Identifier for a project.

Field name	Type	Description
<code>name</code>	string	Required. Name of the project.

`streamSpec`

**Type:** `streamSpecDataObj`

Specification for a stream.

Field name	Type	Description
<code>allowCommitWithoutPassword</code>	boolean	Boolean that indicates whether Coverity Connect will accept new commits of analysis results (through <code>cov-commit-defects</code> ) to the stream without requiring a password. When creating or updating a stream, check your company's security policy before setting to <i>true</i> . If set to <i>true</i> , any user can commit analysis results to the stream.
<code>analysisVersionOverride</code>	string	Specifies the updated Coverity Analysis version that may retrieve data from this stream.

Field name	Type	Description
autoDeleteOnExpiry	boolean	Set to <i>true</i> if the stream should be deleted after a period of inactivity. The default period is 28 days. (Note that the <i>stream.expiration.inactivity.days</i> period is configurable through the <i>cim.properties</i> file. The stream will not be deleted unless it contains at least one snapshot.) Defaults to <i>false</i> with <code>createStream()</code> and <code>createStreamInProject()</code> .
componentMapId	componentMapIdDataObj	Name of the component map with which to associate the stream. Defaults to the <i>Default Component Map</i> with <code>createStream()</code> and <code>createStreamInProject()</code> .
description	string	Description of the stream.
enableDesktopAnalysis	boolean	If true, this stream is able to accept analysis summary information during commit.
language	string	The programming language of the source code files associated with the new stream. Required with <code>createStream()</code> and <code>createStreamInProject()</code> .
name	string	Name of the new stream. Required with <code>createStream()</code> and <code>createStreamInProject()</code> .
outdated	boolean	If true, the filter applies to streams that have been designated as <i>outdated</i> and are hidden from the Coverity Connect UI. If false, the filter applies to non-outdated streams.
ownerAssignmentOption	string	
pluginVersionOverride	string	Specifies the updated Coverity Desktop plug-in version that may retrieve data from this stream.
roleAssignments	roleAssignmentDataObj	Role to associate with the new stream. By default, the username of the stream creator is assigned the <i>streamOwner</i> role for the new stream. See <code>getAllRoles()</code> , <code>getRole()</code> , and <code>getAllPermissions()</code> . If updating role

Field name	Type	Description
summaryExpirationDays	int	assignments, respecify any that you want to retain. Specifies the number of days analysis summaries remain in this stream before being automatically removed from their relative snapshot. If you do not want summaries to be deleted, set summaryExpirationDays to 0. This will reset the value to null, and analysis summaries will not expire.
trriageStoreId	trriageStoreIdDataObj	Identifier for the triage store to associate with the new stream. Required with createStream()and createStreamInProject(). Maximum of one triage store name specification allowed.
versionMismatchMessage	string	Provides additional instructions to analysis users that are using an incorrect version of Coverity Analysis or the Coverity Desktop plug-in.

#### 3.5.11.4. Remarks

See also createStream().

### 3.5.12. Operation: createTriageStore

#### 3.5.12.1. Name

createTriageStore

#### 3.5.12.2. Description

Create a triage store.

#### 3.5.12.3. Parameters

trriageStoreSpec

**Type:** triageStoreSpecDataObj

Field name	Type	Description
description	string	Description of the triage store.
name	string	Name of the triage store. Required with createTriageStore().

Field name	Type	Description
roleAssignments	roleAssignmentDataObj	Role to associate with the triage store at the global level. See getAllRoles(), getRole(), and getAllPermissions(). Zero or more role associations are allowed. If updating role assignments, respecify any that you want to retain.

### 3.5.13. Operation: createUser

#### 3.5.13.1. Name

createUser

#### 3.5.13.2. Description

Create a user.

#### 3.5.13.3. Parameters

userSpec

**Type:** userSpecDataObj

Field name	Type	Description
disabled	boolean	To prevent the user from logging in to the UI, set to <i>true</i> . Defaults to <i>false</i> for createUser().
domain	serverDomainIdDataObj	For an LDAP user only, the LDAP domain of user.
email	string	Email address of the user.
familyName	string	Last name of the user.
givenName	string	First name of the user.
groupNames	groupIdDataObj	Name of an existing group to which the user should belong. Zero or more group associations allowed. <i>Supported when updating a user, not when creating a user.</i>
local	boolean	If an LDAP user, set to <i>false</i> . Defaults to <i>true</i> with createUser(), specifying a local (non-LDAP) user.
locale	string	The locale of the user. Defaults to <i>en-US</i> with createUser().
locked	boolean	To lock out the new user, set to <i>true</i> . Unless password recovery is enabled

Field name	Type	Description
password	string	through the UI, the administrator must reset the password for the locked out user before the user can log in. Defaults to <i>false</i> with <code>createUser()</code> . Password for the user. Required with <code>createUser()</code> .
roleAssignments	roleAssignmentDataObj	Role to associate with the new user at the global level. See <code>getAllRoles()</code> , <code>getRole()</code> , and <code>getAllPermissions()</code> . Zero or more role associations are allowed. If updating role assignments, respecify any that you want to retain.
username	string	Required. Username for/of the user. Any capitalized (upper case) letters in the name will be converted to lower case when using <code>createUser()</code> . Maximum of one name specification is allowed.

### 3.5.14. Operation: deleteAttribute

#### 3.5.14.1. Name

deleteAttribute

#### 3.5.14.2. Description

Delete an attribute.

#### 3.5.14.3. Parameters

attributeDefinitionId

**Type:** attributeDefinitionIdDataObj

Field name	Type	Description
name	string	Required. Name of the attribute.

#### 3.5.14.4. Remarks

To retrieve a list of attributes, see `getAttributes()`.

### 3.5.15. Operation: deleteComponentMap

#### 3.5.15.1. Name

deleteComponentMap

### 3.5.15.2. Description

Delete a component map.

### 3.5.15.3. Parameters

componentMapId

**Type:** componentMapIdDataObj

Field name	Type	Description
name	string	Required. Name of the component map to delete or update.

### 3.5.15.4. Remarks

To retrieve a list of component maps, see `getComponentMaps()`.

## 3.5.16. Operation: deleteGroup

### 3.5.16.1. Name

deleteGroup

### 3.5.16.2. Description

Delete a user group.

### 3.5.16.3. Parameters

groupId

**Type:** groupIdDataObj

Field name	Type	Description
displayName	string	The name of a user group. To retrieve an LDAP group, you use <code>&lt;groupname&gt;@&lt;LDAPserver&gt;</code> .
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group.
name	string	Required. Name of the local or LDAP group.

### 3.5.16.4. Remarks

To retrieve a list of groups, see `getGroups()`.

### 3.5.17. Operation: deleteLdapConfiguration

#### 3.5.17.1. Name

deleteLdapConfiguration

#### 3.5.17.2. Description

Delete an LDAP configuration.

#### 3.5.17.3. Parameters

serverDomainIdDataObj

**Type:** serverDomainIdDataObj

Field name	Type	Description
name	string	Name of the LDAP server domain.

### 3.5.18. Operation: deleteProject

#### 3.5.18.1. Name

deleteProject

#### 3.5.18.2. Description

Delete a project.

#### 3.5.18.3. Parameters

projectId

**Type:** projectIdDataObj

Field name	Type	Description
name	string	Required. Name of the project.

#### 3.5.18.4. Remarks

To get a list of projects, see getProjects().

### 3.5.19. Operation: deleteRole

#### 3.5.19.1. Name

deleteRole

### 3.5.19.2. Description

Delete a role.

### 3.5.19.3. Parameters

roleId

**Type:** roleIdDataObj

Field name	Type	Description
name	string	Required. Name of the role.

### 3.5.19.4. Remarks

To retrieve a list of roles, see getAllRoles().

## 3.5.20. Operation: deleteSnapshot

### 3.5.20.1. Name

deleteSnapshot

### 3.5.20.2. Description

Delete a snapshot.

### 3.5.20.3. Parameters

snapshotId

**Type:** snapshotIdDataObj

Field name	Type	Description
id	long	Numeric identifier for the snapshot. Required.

### 3.5.20.4. Remarks

You need to check on the status of the deletion after making the request. See getDeleteSnapshotJobInfo().

This operation is asynchronous. The snapshot deletion process might take some time for a large snapshot, but the Web Service invocation returns quickly, while the snapshot deletion process continues in the background. If you want to script multiple snapshot deletions from the same stream, keep in mind that while a snapshot is being deleted from the stream, subsequent invocations of this operation, with the original process still running, will fail with a SOAP Fault with error code 1500 and display the following

message: Another process has locked the channel {channel name}. Only a single process may commit data to or delete snapshots from a channel at a time. Please retry later.

### 3.5.21. Operation: deleteStream

#### 3.5.21.1. Name

deleteStream

#### 3.5.21.2. Description

Delete a stream.

#### 3.5.21.3. Parameters

streamId

**Type:** streamIdDataObj

Field name	Type	Description
name	string	Required. Name of the stream.

onlyIfEmpty

**Type:** boolean

### 3.5.22. Operation: deleteTriageStore

#### 3.5.22.1. Name

deleteTriageStore

#### 3.5.22.2. Description

Delete a triage store to which no streams are associated.

#### 3.5.22.3. Parameters

triageStoreId

**Type:** triageStoreIdDataObj

Field name	Type	Description
name	string	Required. Name of the triage store.

#### 3.5.22.4. Remarks

If any streams are associated with the triage store, you must dissociate them from the before you can successfully delete the store. See updateTriageStore().

### **3.5.23. Operation: deleteUser**

#### **3.5.23.1. Name**

deleteUser

#### **3.5.23.2. Description**

Delete a user.

#### **3.5.23.3. Parameters**

username

**Type:** string

### **3.5.24. Operation: executeNotification**

#### **3.5.24.1. Name**

executeNotification

#### **3.5.24.2. Description**

Initiates a view notification to preconfigured recipients. The view must belong to the user who calls this operation.

#### **3.5.24.3. Parameters**

viewname

**Type:** string

#### **3.5.24.4. Remarks**

For details, see the "Email notifications" section of the *Coverity Platform Use and Administration Guide*.

### **3.5.25. Operation: getAllLdapConfigurations**

#### **3.5.25.1. Name**

getAllLdapConfigurations

#### **3.5.25.2. Description**

Retrieve all LDAP configurations.

#### **3.5.25.3. Output (Literal)**

The output of this operation is the argument getAllLdapConfigurationsResponse having the structure defined by the following table.

Name	Type
return	IdapConfigurationDataObj

### 3.5.26. Operation: getAllPermissions

#### 3.5.26.1. Name

getAllPermissions

#### 3.5.26.2. Description

Retrieve the complete list of permissions that can be associated with a role.

#### 3.5.26.3. Output (Literal)

The output of this operation is the argument getAllPermissionsResponse having the structure defined by the following table.

Name	Type
return	permissionDataObj

### 3.5.27. Operation: getAllRoles

#### 3.5.27.1. Name

getAllRoles

#### 3.5.27.2. Description

Retrieve a list of all roles.

#### 3.5.27.3. Output (Literal)

The output of this operation is the argument getAllRolesResponse having the structure defined by the following table.

Name	Type
return	roleDataObj

### 3.5.28. Operation: getArchitectureAnalysisConfiguration

#### 3.5.28.1. Name

getArchitectureAnalysisConfiguration

#### 3.5.28.2. Description

Returns a string carrying configuration information for use with the Architecture Analysis product. It is only intended for internal use by that product.

### 3.5.28.3. Output (Literal)

The output of this operation is the argument `getArchitectureAnalysisConfigurationResponse` having the structure defined by the following table.

Name	Type
return	string

### 3.5.29. Operation: `getAttribute`

#### 3.5.29.1. Name

`getAttribute`

#### 3.5.29.2. Description

Retrieve the properties of a specified attribute.

#### 3.5.29.3. Parameters

`attributeDefinitionId`

**Type:** `attributeDefinitionIdDataObj`

Field name	Type	Description
<code>name</code>	string	Required. Name of the attribute.

#### 3.5.29.4. Output (Literal)

The output of this operation is the argument `getAttributeResponse` having the structure defined by the following table.

Name	Type
return	<code>attributeDefinitionDataObj</code>

### 3.5.30. Operation: `getAttributes`

#### 3.5.30.1. Name

`getAttributes`

#### 3.5.30.2. Description

Retrieve a list of all attributes.

#### 3.5.30.3. Output (Literal)

The output of this operation is the argument `getAttributesResponse` having the structure defined by the following table.

Name	Type
return	attributeDefinitionDataObj

### 3.5.31. Operation: getBackupConfiguration

#### 3.5.31.1. Name

getBackupConfiguration

#### 3.5.31.2. Description

Retrieve the schedule for automated backup of the Coverity Connect database.

#### 3.5.31.3. Output (Literal)

The output of this operation is the argument getBackupConfigurationResponse having the structure defined by the following table.

Name	Type
return	backupConfigurationDataObj

### 3.5.32. Operation: getCategoryNames

#### 3.5.32.1. Name

getCategoryNames

#### 3.5.32.2. Description

Retrieve all known checker category names, as localizedValueDataObj.

#### 3.5.32.3. Output (Literal)

The output of this operation is the argument getCategoryNamesResponse having the structure defined by the following table.

Name	Type
return	localizedValueDataObj

### 3.5.33. Operation: getCheckerNames

#### 3.5.33.1. Name

getCheckerNames

#### 3.5.33.2. Description

Retrieve all known checker names.

### 3.5.33.3. Output (Literal)

The output of this operation is the argument getCheckerNamesResponse having the structure defined by the following table.

Name	Type
return	string

### 3.5.34. Operation: getCommitState

#### 3.5.34.1. Name

getCommitState

#### 3.5.34.2. Description

Find out whether the database will accept new commits of analysis results.

#### 3.5.34.3. Output (Literal)

The output of this operation is the argument getCommitStateResponse having the structure defined by the following table.

Name	Type
return	commitStateDataObj

#### 3.5.34.4. Remarks

See setAcceptingNewCommits().

### 3.5.35. Operation: GetComponent

#### 3.5.35.1. Name

GetComponent

#### 3.5.35.2. Description

Retrieve the properties of a component.

#### 3.5.35.3. Parameters

componentId

**Type:** componentIdDataObj

Field name	Type	Description
name	string	Required. Name of a component in the form

Field name	Type	Description
		<b>componentMapName.componentName</b> (for example, myComponentMap.myComponent).

#### 3.5.35.4. Output (Literal)

The output of this operation is the argument `GetComponentResponse` having the structure defined by the following table.

Name	Type
return	componentDataObj

#### 3.5.36. Operation: `GetComponentMaps`

##### 3.5.36.1. Name

`GetComponentMaps`

##### 3.5.36.2. Description

Retrieve a list of component maps that matches a component name pattern.

##### 3.5.36.3. Parameters

`filterSpec`

**Type:** `componentMapFilterSpecDataObj`

Field name	Type	Description
<code>namePattern</code>	string	Glob pattern matching the name of one or more component maps.

#### 3.5.36.4. Output (Literal)

The output of this operation is the argument `GetComponentMapsResponse` having the structure defined by the following table.

Name	Type
return	componentMapDataObj

#### 3.5.37. Operation: `getDefectStatuses`

##### 3.5.37.1. Name

`getDefectStatuses`

### 3.5.37.2. Description

Retrieve the list of Status attribute values that can be associated with a software issue.

### 3.5.37.3. Output (Literal)

The output of this operation is the argument getDefectStatusesResponse having the structure defined by the following table.

Name	Type
return	string

### 3.5.38. Operation: getDeleteSnapshotJobInfo

#### 3.5.38.1. Name

getDeleteSnapshotJobInfo

#### 3.5.38.2. Description

Find out whether a snapshot deletion process succeeded.

#### 3.5.38.3. Parameters

snapshotId

**Type:** snapshotIdDataObj

Field name	Type	Description
id	long	Numeric identifier for the snapshot. Required.

#### 3.5.38.4. Output (Literal)

The output of this operation is the argument getDeleteSnapshotJobInfoResponse having the structure defined by the following table.

Name	Type
return	deleteSnapshotJobInfoDataObj

#### 3.5.38.5. Remarks

You need to call this operation after calling deleteSnapshot().

### 3.5.39. Operation: getDeveloperStreamsProjects

#### 3.5.39.1. Name

getDeveloperStreamsProjects

### 3.5.39.2. Description

Get a list of project specifications in developer streams (for all such projects or for a filtered set of such projects).

### 3.5.39.3. Parameters

filterSpec

**Type:** projectFilterSpecDataObj

Field name	Type	Description
descriptionPattern	string	Glob pattern matching the description of one or more projects.
includeChildren	boolean	Value of <i>false</i> if the results <i>should not</i> include roles and other properties associated with the project. Defaults to <i>true</i> .
includeStreams	boolean	Value of <i>false</i> if the results <i>should not</i> include streams associated with the project. Defaults to <i>true</i> .
namePattern	string	Glob pattern matching the name of one or more projects.

### 3.5.39.4. Output (Literal)

The output of this operation is the argument getProjectsResponse having the structure defined by the following table.

Name	Type
return	projectDataObj

### 3.5.39.5. Remarks

When the name and description filters are both specified, both patterns must match the project for the project to return in the resulting list.

## 3.5.40. Operation: getGroup

### 3.5.40.1. Name

getGroup

### 3.5.40.2. Description

Retrieve the properties of a user group.

### 3.5.40.3. Parameters

groupId

**Type:** groupIdDataObj

Specification for a user group identifier.

Field name	Type	Description
displayName	string	The name of a user group. To retrieve an LDAP group, you use <groupname>@<LDAPserver>.
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group.
name	string	Required. Name of the local or LDAP group.

### 3.5.40.4. Output (Literal)

The output of this operation is the argument getGroupResponse having the structure defined by the following table.

Name	Type
return	groupDataObj

## 3.5.41. Operation: getGroups

### 3.5.41.1. Name

getGroups

### 3.5.41.2. Description

Get a list of groups.

### 3.5.41.3. Parameters

filterSpec

**Type:** groupFilterSpecDataObj

Field name	Type	Description
ldap	boolean	Value of <i>true</i> for LDAP groups only; otherwise, <i>false</i> .
namePattern	string	Glob pattern matching the name of one or more groups.

Field name	Type	Description
projectIdDataObj	projectIdDataObj	Name of a project with which the group must have a role association.
userList	string	User name of a user that must belong to the group. Multiple names allowed.

pageSpec

**Type:** pageSpecDataObj

Field name	Type	Description
pageSize	int	Required. Number of records to return. Maximum of 1000.
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Do not specify with <code>getGroups()</code> .  Name of the field to use for sorting results. For example, the <code>dateCreated</code> or <code>familyName</code> field for user records returned by <code>getUsers()</code> . You cannot sort by a field that can appear multiple times for a single record (for example, a single user record can have multiple group fields).  For examples of <code>sortField</code> values, see the remark for this data object.
startIndex	int	Zero-based index of user records to return. Defaults to <i>0</i> . You might use this field if there are more than 1000 users. You could make separate calls that start the index at 1000 or 2000, for example, until you capture separate lists of all of the users.

#### 3.5.41.4. Output (Literal)

The output of this operation is the argument `getGroupsResponse` having the structure defined by the following table.

**Note:** The object that is returned by this operation can include `roleAssignments` that are identical. This occurs because `roleAssignment` objects don't have an attribute that specifies a name for the type to which they apply (for example, a project name).

Name	Type
return	groupsPageDataObj

### 3.5.42. Operation: getLdapServerDomains

#### 3.5.42.1. Name

getLdapServerDomains

#### 3.5.42.2. Description

Retrieves the host name or host IP of one or more LDAP servers.

#### 3.5.42.3. Output (Literal)

The output of this operation is the argument getLdapServerDomainsResponse having the structure defined by the following table.

Name	Type
return	serverDomainIdDataObj

#### 3.5.42.4. Remarks

See createLdapConfiguration().

### 3.5.43. Operation: getLicenseConfiguration

#### 3.5.43.1. Name

getLicenseConfiguration

#### 3.5.43.2. Description

Retrieve details on your Coverity Connect license.

#### 3.5.43.3. Output (Literal)

The output of this operation is the argument getLicenseConfigurationResponse having the structure defined by the following table.

Name	Type
return	licenseDataObj

#### 3.5.43.4. Remarks

If you think that the license data is incorrect, or if you want more information on obtaining a new license, contact Synopsys sales support at [software-integrity-support@synopsys.com](mailto:software-integrity-support@synopsys.com).

### 3.5.44. Operation: **getLicenseState**

#### 3.5.44.1. Name

getLicenseState

#### 3.5.44.2. Description

Find out whether Coverity Desktop analysis is enabled through your license.

#### 3.5.44.3. Output (Literal)

The output of this operation is the argument getLicenseStateResponse having the structure defined by the following table.

Name	Type
return	licenseStateDataObj

### 3.5.45. Operation: **getLoggingConfiguration**

#### 3.5.45.1. Name

getLoggingConfiguration

#### 3.5.45.2. Description

Retrieve information about your current Coverity Connect configurations for logging.

#### 3.5.45.3. Output (Literal)

The output of this operation is the argument getLoggingConfigurationResponse having the structure defined by the following table.

Name	Type
return	loggingConfigurationDataObj

#### 3.5.45.4. Remarks

See also setLoggingConfiguration().

### 3.5.46. Operation: **getMessageOfTheDay**

#### 3.5.46.1. Name

getMessageOfTheDay

#### 3.5.46.2. Description

Get the message of the day.

### 3.5.46.3. Output (Literal)

The output of this operation is the argument `getMessageOfTheDayResponse` having the structure defined by the following table.

Name	Type
return	string

### 3.5.46.4. Remarks

See `setMessageOfTheDay()`.

## 3.5.47. Operation: `getOutputFileForSnapshot`

### 3.5.47.1. Name

`getOutputFileForSnapshot`

### 3.5.47.2. Description

Get the specified output file for the specified snapshot. Output files are the files that Coverity Analysis writes to the intermediate directory's `output` directory.

### 3.5.47.3. Parameters

`snapshotId`

**Type:** `snapshotIdDataObj`

Identifies the snapshot whose output file you want to retrieve.

Field name	Type	Description
<code>id</code>	<code>long</code>	Numeric identifier for the snapshot. Required.

`fileName`

**Type:** `string`

Identifies the output file to retrieve.

### 3.5.47.4. Output (Literal)

The output of this operation is the argument `getOutputFileForSnapshotResponse` of type `getOutputFileForSnapshotResponse` having the structure defined by the following table.

Name	Type
return	<code>outputFileDataObj</code>

### 3.5.48. Operation: getProjects

#### 3.5.48.1. Name

getProjects

#### 3.5.48.2. Description

Get a list of project specifications (for all projects or for a filtered set of projects).

#### 3.5.48.3. Parameters

filterSpec

**Type:** projectFilterSpecDataObj

Field name	Type	Description
descriptionPattern	string	Glob pattern matching the description of one or more projects.
includeChildren	boolean	Value of <i>false</i> if the results <i>should not</i> include roles and other properties associated with the project. Defaults to <i>true</i> .
includeStreams	boolean	Value of <i>false</i> if the results <i>should not</i> include streams associated with the project. Defaults to <i>true</i> .
namePattern	string	Glob pattern matching the name of one or more projects.

#### 3.5.48.4. Output (Literal)

The output of this operation is the argument getProjectsResponse having the structure defined by the following table.

Name	Type
return	projectDataObj

#### 3.5.48.5. Remarks

When the name and description filters are both specified, both patterns must match the project for the project to return in the resulting list.

### 3.5.49. Operation: getRole

#### 3.5.49.1. Name

getRole

### 3.5.49.2. Description

Retrieve the properties of a role, including its associated permissions.

### 3.5.49.3. Parameters

roleId

**Type:** roleIdDataObj

Field name	Type	Description
name	string	Required. Name of the role.

### 3.5.49.4. Output (Literal)

The output of this operation is the argument getRoleResponse having the structure defined by the following table.

Name	Type
return	roleDataObj

## 3.5.50. Operation: getServerTime

### 3.5.50.1. Name

getServerTime

### 3.5.50.2. Description

Retrieves the current date and time from the server.

### 3.5.50.3. Output (Literal)

The output of this operation is the argument getServerTimeResponse having the structure defined by the following table.

Name	Type
return	dateTime

## 3.5.51. Operation: getSignInConfiguration

### 3.5.51.1. Name

getSignInConfiguration

### 3.5.51.2. Description

Retrieve sign-in settings. These configurations are identical to Sign In Settings (a set of System Configuration settings) in the Coverity Connect UI.

### 3.5.51.3. Output (Literal)

The output of this operation is the argument `getSignInConfigurationResponse` having the structure defined by the following table.

Name	Type
return	signInSettingsDataObj

### 3.5.52. Operation: `getSkeletonizationConfiguration`

#### 3.5.52.1. Name

`getSkeletonizationConfiguration`

#### 3.5.52.2. Description

Retrieve the configuration for the process that purges snapshot details. Purging these details can help you reduce and maintain the database size.

#### 3.5.52.3. Output (Literal)

The output of this operation is the argument `getSkeletonizationConfigurationResponse` having the structure defined by the following table.

Name	Type
return	skeletonizationConfigurationDataObj

#### 3.5.52.4. Remarks

See also `setSkeletonizationConfiguration()`.

### 3.5.53. Operation: `getSnapshotInformation`

#### 3.5.53.1. Name

`getSnapshotInformation`

#### 3.5.53.2. Description

Retrieve information about a snapshot in a stream.

#### 3.5.53.3. Parameters

`snapshotIds`

**Type:** `snapshotIdDataObj`

Identifier for a snapshot.

Field name	Type	Description
id	long	Numeric identifier for the snapshot. Required.

#### 3.5.53.4. Output (Literal)

The output of this operation is the argument `getSnapshotInformationResponse` having the structure defined by the following table.

Name	Type
return	snapshotInfoDataObj

#### 3.5.53.5. Remarks

To retrieve a snapshot ID, see `getSnapshotForStream()`. The ID is also available through the Coverity Connect UI.

### 3.5.54. Operation: `getSnapshotPurgeDetails`

#### 3.5.54.1. Name

`getSnapshotPurgeDetails`

#### 3.5.54.2. Description

**Deprecated in v8:** Use `getSkeletonizationConfiguration()` instead to retrieve the configuration for the process that purges snapshot details. Purging these details can help you reduce and maintain the database size.

#### 3.5.54.3. Output (Literal)

The output of this operation is the argument `getSnapshotPurgeDetailsResponse` having the structure defined by the following table.

Name	Type
return	snapshotPurgeDetailsObj

### 3.5.55. Operation: `getSnapshotsForStream`

#### 3.5.55.1. Name

`getSnapshotsForStream`

#### 3.5.55.2. Description

Retrieve a set of snapshots that belong to a specified stream.

### 3.5.55.3. Parameters

streamId

**Type:** streamIdDataObj

Field name	Type	Description
name	string	Required. Name of the stream.

filterSpec

**Type:** snapshotFilterSpecDataObj

Field name	Type	Description
descriptionPattern	string	Glob pattern matching the description of one or more snapshots.
endDate	dateTime	Date (and optionally, time) on or before which the snapshot was created. Serves as an upper bound on the IDs to return. (If you do not enter a time, the system is likely to assume 12:00 a.m.) See the sample request below.
hasSummaries	boolean	If present, only snapshots with a hasSummaries attribute that is equal to the specified value will be returned. If absent, no filtering on hasSummaries takes place.
lastBeforeCodeVersionDate	dateTime	If present, only one snapshot will be returned, specifically, the snapshot with the latest codeVersionDate among those that are before or equal to the specified date. If there is no such snapshot, then the call will return an empty set.
startDate	dateTime	Date (and optionally, time) on or after when the snapshot was created. Serves as a lower bound on the IDs to return. (If you do not enter a time, the system is likely to assume 12:00 a.m.) See the sample request below.
targetPattern	string	Glob pattern matching the target of the snapshot.
versionPattern	string	Glob pattern matching the version of the snapshot.

### 3.5.55.4. Output (Literal)

The output of this operation is the argument getSnapshotsForStreamResponse having the structure defined by the following table.

Name	Type
return	snapshotIdDataObj

### 3.5.56. Operation: getStandardAttribute

#### 3.5.56.1. Name

getStandardAttribute

#### 3.5.56.2. Description

Retrieve the properties of a specified standard attribute.

#### 3.5.56.3. Parameters

standardAttributeIdDataObj

**Type:** standardAttributeIdDataObj

Required. Name of the standard attribute.

Field name	Type	Description
name	string	Required. Name of the standard attribute.

#### 3.5.56.4. Output (Literal)

The output of this operation is the argument getStandardAttributeResponse having the structure defined by the following table.

Name	Type
return	standardAttributeDataObj

### 3.5.57. Operation: getStandardAttributes

#### 3.5.57.1. Name

getStandardAttributes

#### 3.5.57.2. Description

Retrieve a list of all standard attributes.

### 3.5.57.3. Output (Literal)

The output of this operation is the argument `getStandardAttributesResponse` having the structure defined by the following table.

Name	Type
return	standardAttributeDataObj

### 3.5.57.4. Remarks

To retrieve a snapshot ID, see `getSnapshotForStream()`. The ID is also available through the Coverity Connect UI.

## 3.5.58. Operation: `getStreams`

### 3.5.58.1. Name

`getStreams`

### 3.5.58.2. Description

Retrieve a set of streams.

### 3.5.58.3. Parameters

`filterSpec`

**Type:** `streamFilterSpecDataObj`

Field name	Type	Description
<code>languageList</code>	string	Programming language matching that of one or more streams. Zero or more language filters allowed.
<code>descriptionPattern</code>	string	Glob pattern matching the description of one or more streams.
<code>namePattern</code>	string	Glob pattern matching the name of one or more streams.

### 3.5.58.4. Output (Literal)

The output of this operation is the argument `getStreamsResponse` having the structure defined by the following table.

Name	Type
return	streamDataObj

### 3.5.59. Operation: `getSystemConfig`

#### 3.5.59.1. Name

`getSystemConfig`

#### 3.5.59.2. Description

Retrieves system configuration properties, including properties of the database with which this API communicates.

#### 3.5.59.3. Output (Literal)

The output of this operation is the argument `getSystemConfigResponse` having the structure defined by the following table.

Name	Type
return	<code>configurationDataObj</code>

### 3.5.60. Operation: `getTriageStores`

#### 3.5.60.1. Name

`getTriageStores`

#### 3.5.60.2. Description

Retrieve a set of triage store specifications, including stream associations.

#### 3.5.60.3. Parameters

`filterSpec`

**Type:** `trriageStoreFilterSpecDataObj`

Field name	Type	Description
<code>descriptionPattern</code>	string	Glob pattern matching the description of one or more triage stores.
<code>namePattern</code>	string	Glob pattern matching one or more names of the set of triage stores to retrieve.

#### 3.5.60.4. Output (Literal)

The output of this operation is the argument `getTriageStoresResponse` having the structure defined by the following table.

Name	Type
return	<code>trriageStoreDataObj</code>

### 3.5.61. Operation: getTypeNames

#### 3.5.61.1. Name

getTypeNames

#### 3.5.61.2. Description

Returns a list of all known defect types, as localizedValueDataObj.

#### 3.5.61.3. Output (Literal)

The output of this operation is the argument getTypeNamesResponse having the structure defined by the following table.

Name	Type
return	localizedValueDataObj

### 3.5.62. Operation: getUser

#### 3.5.62.1. Name

getUser

#### 3.5.62.2. Description

Retrieve a user by user name.

#### 3.5.62.3. Parameters

username

**Type:** string

User name.

#### 3.5.62.4. Output (Literal)

The output of this operation is the argument getUserResponse having the structure defined by the following table.

Name	Type
return	userDataObj

### 3.5.63. Operation: getUsers

#### 3.5.63.1. Name

getUsers

### 3.5.63.2. Description

Get users (filtered or unfiltered).

### 3.5.63.3. Parameters

filterSpec

**Type:** userFilterSpecDataObj

Field name	Type	Description
assignable	boolean	Set to <i>true</i> to retrieve only those users who can own software issues; <i>false</i> to retrieve only those who cannot. Otherwise, do not set.
disabled	boolean	Set to <i>true</i> to retrieve <i>disabled</i> users only. Set to <i>false</i> to retrieve enabled users only. Otherwise, do not set.
groupsList	string	Name of user group to which the retrieved users must belong. Zero or more groups allowed.
includeDetails	boolean	Set to <i>false</i> to prevent the inclusion of user details in the response. Defaults to <i>true</i> .
ldap	boolean	Set to <i>true</i> to retrieve only LDAP users; <i>false</i> to retrieve only local users. Otherwise, do not set.
locked	boolean	Set to <i>true</i> to retrieve only those users who have been locked out; <i>false</i> to retrieve only unlocked users. Otherwise, do not set.
namePattern	string	Glob pattern that matches the user name of the users to retrieve.
projectIdDataObj	projectIdDataObj	Name of project to which the retrieved set of users must have a role association.
startId	long	Internal. Do not specify.

pageSpec

**Type:** pageSpecDataObj

Field name	Type	Description
pageSize	int	Required. Number of records to return. Maximum of 1000.

Field name	Type	Description
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Do not specify with <code>getGroups()</code> .  Name of the field to use for sorting results. For example, the <code>dateCreated</code> or <code>familyName</code> field for user records returned by <code>getUsers()</code> . You cannot sort by a field that can appear multiple times for a single record (for example, a single user record can have multiple group fields).  For examples of <code>sortField</code> values, see the remark for this data object.
startIndex	int	Zero-based index of user records to return. Defaults to <i>0</i> . You might use this field if there are more than 1000 users. You could make separate calls that start the index at 1000 or 2000, for example, until you capture separate lists of all of the users.

#### 3.5.63.4. Output (Literal)

The output of this operation is the argument `getUsersResponse` having the structure defined by the following table.

Name	Type
return	<code>usersPageDataObj</code>

#### 3.5.64. Operation: `getVersion`

##### 3.5.64.1. Name

`getVersion`

##### 3.5.64.2. Description

Retrieve the version of Coverity Connect.

##### 3.5.64.3. Output (Literal)

The output of this operation is the argument `getVersionResponse` having the structure defined by the following table.

Name	Type
return	versionDataObj

### 3.5.65. Operation: importLicense

#### 3.5.65.1. Name

importLicense

#### 3.5.65.2. Description

Import your Coverity Connect license file, license.dat.

#### 3.5.65.3. Parameters

licenseSpecDataObj

**Type:** licenseSpecDataObj

Field name	Type	Description
licenseDataFile	string	Contents of the license file. See the example for details.

### 3.5.66. Operation: notify

#### 3.5.66.1. Name

notify

#### 3.5.66.2. Description

Send an email notification to a specified user.

#### 3.5.66.3. Parameters

usernames

**Type:** string

One or more usernames.

subject

**Type:** string

Subject-line text for the email.

message

**Type:** string

Body text for the email.

### 3.5.66.4. Output (Literal)

The output of this operation is the argument notifyResponse having the structure defined by the following table.

Name	Type
return	string

### 3.5.66.5. Remarks

Precondition to using this operation: Coverity Connect must be properly configured to send email.

## 3.5.67. Operation: refreshLdapGroup

### 3.5.67.1. Name

refreshLdapGroup

### 3.5.67.2. Description

Refresh an LDAP group.

### 3.5.67.3. Parameters

groupId

**Type:** groupIdDataObj

Specification for a user group identifier.

Field name	Type	Description
displayName	string	The name of a user group. To retrieve an LDAP group, you use <groupname>@<LDAPserver>.
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group.
name	string	Required. Name of the local or LDAP group.

### 3.5.67.4. Output

None

## 3.5.68. Operation: setAcceptingNewCommits

### 3.5.68.1. Name

setAcceptingNewCommits

### 3.5.68.2. Description

Control whether the database will accept new commits of analysis results.

### 3.5.68.3. Parameters

acceptNewCommits

**Type:** boolean

### 3.5.68.4. Remarks

See getCommitState().

## 3.5.69. Operation: setArchitectureAnalysisConfiguration

### 3.5.69.1. Name

setArchitectureAnalysisConfiguration

### 3.5.69.2. Description

Set the value that getArchitectureAnalysisConfiguration returns.

### 3.5.69.3. Parameters

architectureAnalysisConfiguration

**Type:** string

## 3.5.70. Operation: setBackupConfiguration

### 3.5.70.1. Name

setBackupConfiguration

### 3.5.70.2. Description

Set a schedule for automated backup of the Coverity Connect database. The name of the backup file looks something like the following: CIM.2013-12-04.10-35.backup

### 3.5.70.3. Parameters

backupConfigurationDataObj

**Type:** backupConfigurationDataObj

The settings correspond to the Maintenance options (System settings) in Coverity Connect.

Field name	Type	Description
backupLocation	string	Full path to a backup directory on the disk where Coverity Connect is installed.

Field name	Type	Description
backupTime	string	Time of the backup, for example: 01:15
fridayEnabled	boolean	Friday backup if set to true. Defaults to false.
mondayEnabled	boolean	Monday backup if set to true. Defaults to false.
saturdayEnabled	boolean	Saturday backup if set to true. Defaults to false.
sundayEnabled	boolean	Sunday backup if set to true. Defaults to false.
thursdayEnabled	boolean	Thursday backup if set to true. Defaults to false.
tuesdayEnabled	boolean	Tuesday backup if set to true. Defaults to false.
wednesdayEnabled	boolean	Wednesday backup if set to true. Defaults to false.

#### 3.5.70.4. Remarks

See also `getBackupConfiguration()`.

### 3.5.71. Operation: `setLoggingConfiguration`

#### 3.5.71.1. Name

`setLoggingConfiguration`

#### 3.5.71.2. Description

Enable or disable logging options for Coverity Connect.

Coverity Connect automatically logs system event information to the `cim.log` file. You can increase the amount of information that Coverity Connect records to this file by enabling additional logging options to work with Coverity Support on an issue. Coverity recommends that you leave all of the logging options disabled and only enable them by request from Coverity Support.

#### 3.5.71.3. Parameters

`loggingConfigurationDataObj`

**Type:** `loggingConfigurationDataObj`

The settings correspond to the Logging Configuration options (System settings) in Coverity Connect.

Field name	Type	Description
<code>accessControlLogging</code>	boolean	Access control logging.

Field name	Type	Description
backgroundLogging	boolean	Background tasks logging.
bugTrackingSystemLogging	boolean	Bug tracking system logging.
commitLogging	boolean	Commit logging.
configurationLogging	boolean	Configuration logging.
databaseLogging	boolean	Database logging.
frameworkLogging	boolean	Framework logging.
internalLogging	boolean	Internal logging.
kerberosLogging	boolean	Kerberos logging.
metricsAndHistoryLogging	boolean	Metrics and history logging.
notificationLogging	boolean	Notifications logging.
performanceLogging	boolean	Performance logging.
policyManagerLogging	boolean	Policy Manager logging.
remoteConfigLogging	boolean	Remote Configuration logging.
requestPerformanceLogging	boolean	Request Performance logging.
skeletonizationLogging	boolean	Snapshot Details Purge logging.
testAdvisorLogging	boolean	Test Advisor logging.
triageLogging	boolean	Triage logging.
triageSynchLogging	boolean	Triage & CID Synchronization logging.
webLogging	boolean	Web logging.
webServicesLogging	boolean	Web Services logging.

#### 3.5.71.4. Remarks

See also `getLoggingConfiguration()`.

#### 3.5.72. Operation: `setMessageOfTheDay`

##### 3.5.72.1. Name

`setMessageOfTheDay`

##### 3.5.72.2. Description

Set the message of the day.

##### 3.5.72.3. Parameters

message

**Type:** string

Message text.

### 3.5.72.4. Remarks

See `getMessageOfTheDay()`.

## 3.5.73. Operation: `setSkeletonizationConfiguration`

### 3.5.73.1. Name

`setSkeletonizationConfiguration`

### 3.5.73.2. Description

Configure the process that purges snapshot details. Purging these details can help you reduce and maintain the database size.

### 3.5.73.3. Parameters

`skeletonizationConfigurationDataObj`

**Type:** `skeletonizationConfigurationDataObj`

These System settings correspond to the Maintenance options for Snapshot Details Purge in Coverity Connect.

Field name	Type	Description
<code>daysBeforeSkeletonization</code>	int	Age (in number of days) at which the details of a snapshot can be purged.
<code>fridayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>minSnapshotsToKeep</code>	int	Number of snapshots that must retain their details. Minimum of one is required.
<code>mondayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>saturdayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>sundayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>thursdayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>time</code>	string	Time of day at which the purge should take place. Example for 5:00 a.m.: <i>05:00</i>
<code>tuesdayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.

Field name	Type	Description
wednesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.

#### 3.5.73.4. Remarks

See also, `getSkeletonizationConfiguration()`.

### 3.5.74. Operation: `setSnapshotPurgeDetails`

#### 3.5.74.1. Name

`setSnapshotPurgeDetails`

#### 3.5.74.2. Description

**Deprecated in v8:** Use `setSkeletonizationConfiguration()` instead to configure the process that purges snapshot details.

#### 3.5.74.3. Parameters

`purgeDetailsSpec`

**Type:** `snapshotPurgeDetailsObj`

These System settings correspond to the Maintenance options for Snapshot Details Purge in Coverity Connect.

Field name	Type	Description
<code>daysBeforeSkeletonization</code>	int	Age (in number of days) at which the details of a snapshot can be purged. Defaults to 0.
<code>fridayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>minSnapshotsToKeep</code>	int	Number of snapshots that must retain their details. Defaults to 0.
<code>mondayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>saturdayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>sundayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>thursdayEnabled</code>	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
<code>timeOfDay</code>	string	Require Time of day at which the purge should take place. Example for 5:00 a.m.: <i>05:00</i>

Field name	Type	Description
tuesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
wednesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.

#### 3.5.74.4. Remarks

See also, `getSnapshotPurgeDetails()`.

#### 3.5.75. Operation: `updateAttribute`

##### 3.5.75.1. Name

`updateAttribute`

##### 3.5.75.2. Description

Update an attribute specification.

##### 3.5.75.3. Parameters

`attributeDefinitionId`

**Type:** `attributeDefinitionIdDataObj`

Field name	Type	Description
<code>name</code>	<code>string</code>	Required. Name of the attribute.

`attributeDefinitionSpec`

**Type:** `attributeDefinitionSpecDataObj`

Field name	Type	Description
<code>attributeName</code>	<code>string</code>	Name for the attribute. Required when using <code>createAttribute()</code> .
<code>attributeType</code>	<code>string</code>	The type of attribute. Required when using <code>createAttribute()</code> .
<code>attributeValueChangeSpec</code>	<code>attributeValueChangeSpecDataObj</code>	For a LIST_OF_VALUES attribute type only: The set of values available to the attribute.
<code>defaultValue</code>	<code>string</code>	For a LIST_OF_VALUES attribute type only: The default attribute value.
<code>description</code>	<code>string</code>	Description of the attribute.

Field name	Type	Description
showInTriage	boolean	If <i>true</i> , makes the attribute available for use in the Triage pane of the UI.

### 3.5.76. Operation: updateComponentMap

#### 3.5.76.1. Name

updateComponentMap

#### 3.5.76.2. Description

Update one or more properties of a component map.

#### 3.5.76.3. Parameters

componentMapId

**Type:** componentMapIdDataObj

Field name	Type	Description
name	string	Required. Name of the component map to delete or update.

componentMapSpec

**Type:** componentMapSpecDataObj

Field name	Type	Description
componentMapName	string	Required. Name of the component map.
componentPathRules	componentPathRuleDataObj	Pattern matching a path to set of files to associate with a component for the component map. Multiple component map rules allowed.
components	componentDataObj	Specification of the component that you are associating with the componentmap. Multiple component associations allowed. When <i>updating</i> (not creating) components, you must specify the complete list of your components, including the "Other" component because any component in the component map that does not appear in the update list will be deleted. Alternatively, you can pass <b>null</b> to

Field name	Type	Description
		retain all existing components during an update.
defectRules	componentDefectRuleDataObj	Default owner of the specified component. Each component can have an owner.
description	string	Description of the component map.
forceDeleteComponents	boolean	When set to <code>true</code> , components that do not appear in the update list and are chosen to be included in hierarchies will be deleted and removed from the hierarchies. When set to <code>false</code> , components that do not appear in the update list and are chosen to be included in hierarchies will <i>not</i> be deleted. In the latter case, an error message is returned.

---

### 3.5.77. Operation: updateGroup

#### 3.5.77.1. Name

updateGroup

#### 3.5.77.2. Description

Update a group specification.

#### 3.5.77.3. Parameters

groupId

**Type:** groupIdDataObj

Field name	Type	Description
displayName	string	The name of a user group. To retrieve an LDAP group, you use <groupname>@<LDAPserver>.
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group.
name	string	Required. Name of the local or LDAP group.

---

groupSpec

**Type:** groupSpecDataObj

Field name	Type	Description
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group. Required when using createGroup() for an LDAP group. Maximum of one <i>domain</i> specification allowed.
local	boolean	Defaults to <i>true</i> , specifying a local (non-LDAP) group. If an LDAP group, set to <i>false</i> when using createGroup(). Maximum of one <i>local</i> specification allowed.
name	string	Name for the user group. Required when using createGroup().
roleAssignments	roleAssignmentDataObj	Role to associate with the group at the global level. See getAllRoles(), getRole(), and getAllPermissions(). Zero or more role associations allowed. If updating role assignments, respecify any global type roles that you want to retain.
syncEnabled	boolean	If a local group, set to <i>false</i> when using createRole(). Defaults to <i>true</i> . Applies to an LDAP group only.

### 3.5.78. Operation: updateLdapConfiguration

#### 3.5.78.1. Name

updateLdapConfiguration

#### 3.5.78.2. Description

Update an LDAP configuration.

#### 3.5.78.3. Parameters

serverDomainIdDataObj

**Type:** serverDomainIdDataObj

Field name	Type	Description
name	string	Name of the LDAP server domain.

ldapConfigurationSpec

**Type:** ldapConfigurationSpecDataObj

Field name	Type	Description
anonymousBind	boolean	Value of <i>true</i> if using an anonymous bind; otherwise, <i>false</i> to indicate a general bind with a shared bind DN.
baseDN	string	LDAP domain name in host format, such as <domain>.com or <corp.domain>.<com>, or base DN such as <i>ou=corp</i> .
bindName	string	The username needed to access the LDAP server.  You can specify the user in <i>username@domain</i> or LDAP DN format.
bindPassword	string	Bind user password for LDAP user queries. Required for authenticated binding requests. If you do not enter a password, and if your LDAP server is configured to accept unauthenticated binding requests, Coverity Connect will attempt to gain unauthenticated access to the LDAP server. If an unauthenticated LDAP connection fails, Coverity Connect will display an LDAP Server Configuration failed message and an explanation of the failure. You can find more information about the failure in the <i>cim.log</i> file.
displayName	string	Name that identifies the LDAP server in a multi-server setup. Appears in the UI.
groupFilter	string	Optional filter to use when importing LDAP groups. Takes a valid RFC format. Example: <i>cn=eng*</i>
groupFullName	boolean	If <i>false</i> , group members are stored by <i>userName</i> . If <i>true</i> , members are stored by their full DN. Defaults to <i>true</i> .
groupMember	string	LDAP attribute that defines the members of a group. Group members can be referred to by their DN or <i>username</i> .
groupName	string	Name attribute for the group. For Example: <i>cn</i>
groupObjectClass	string	LDAP <i>objectClass</i> value that identifies user groups. For OpenLDAP, the

Field name	Type	Description
		default is <i>groupofnames</i> . For Active Directory, the default is <i>group</i> . This field defines a component of the LDAP user group search query that Coverity Connect creates.
groupSearchBase	string	The group search base DN to prepend to the base DN. Used for group searches. The base DN is configured as <i>Domain</i> as part of basic LDAP settings. For example: <i>cn=groups, ou=corp</i>
primary	boolean	Value of <i>true</i> if this configuration is the primary one. Defaults to <i>false</i> .
secureConnection	boolean	Value of <i>true</i> if using a secure connection. SSL or TLS protocol required. For information about LDAP server requirements, secure connections, importing LDAP users and groups, and other details about LDAP, refer to <i>Coverity Connect Usage and Administration Guide</i> .
serverDomain	string	The host name or host IP of the LDAP server. The name must be resolvable from the Coverity Connect host.
serverPort	long	The TCP port number where the LDAP server listens for connections.
tlsEnabled	boolean	Value of <i>true</i> if TLS security protocol is used.
userEmail	string	LDAP attribute that maps to the email address of the user.
userFirstName	string	LDAP attribute that maps to the first name (given name) of the user.
userLastName	string	LDAP attribute that maps to the last name (surname) of the user.
userName	string	Username for testing user search settings. Maps to the username in the database.
userObjectClass	string	LDAP <i>objectClass</i> associated with users on the LDAP server. For OpenLDAP, the default is <i>inetOrgPerson</i> . For Active Directory, the default is <i>user</i> .

Field name	Type	Description
userSearchBase	string	Relative DN to prepend to base DN to limit a user search. For example, ou=Users will search in ou=Users, dc=ad, dc=coverity, dc=com.

### 3.5.79. Operation: updateProject

#### 3.5.79.1. Name

updateProject

#### 3.5.79.2. Description

Update a project specification.

#### 3.5.79.3. Parameters

projectId

**Type:** projectIdDataObj

Passes the identifier for a project.

Field name	Type	Description
name	string	Required. Name of the project.

projectSpec

**Type:** projectSpecDataObj

Field name	Type	Description
streams	streamIdDataObj	Name of an existing stream to associate with the project. When updating a project, you must list all of the streams that you want to retain except for streams with the __internal prefix to their name, such as Dynamic Analysis streams. You might see such internal streams listed when you call getProjects(). Zero or more stream name associations allowed. See the remark for additional details.
streamLinks	streamIdDataObj	Name of a new or existing stream link to associate with the project. (Note that a stream link and the stream to which it links must belong to separate projects.) Zero or more stream link name associations allowed.

Field name	Type	Description
description	string	Description of the project.
name	string	Name for the project. Required when using createProject().
roleAssignments	roleAssignmentDataObj	Role to associate with the project. See getAllRoles(), getRole(), and getAllPermissions(). By default, the username of the project creator is assigned the <i>projectOwner</i> role for the new stream. If updating role assignments, respecify any that you want to retain. Zero or more role associations allowed.

### 3.5.79.4. Remarks

The streams and streamLinks fields must include the complete list of streams or stream links. Any stream or stream link that appears in the project but does not appear in the corresponding projectSpecDataObj list will be deleted from the project.

### 3.5.80. Operation: updateRole

#### 3.5.80.1. Name

updateRole

#### 3.5.80.2. Description

Update a role specification.

#### 3.5.80.3. Parameters

roleId

**Type:** roleIdDataObj

Field name	Type	Description
name	string	Required. Name of the role.

roleSpec

**Type:** roleSpecDataObj

Field name	Type	Description
deletable	boolean	If the new role can be deleted, set to <i>true</i> . If not, set to <i>false</i> . Required when using createRole(). Maximum of one specification allowed.
description	string	Description of the role.

Field name	Type	Description
editable	boolean	Value of <i>false</i> not allowed. Requires value of <i>true</i> when using createRole(). Maximum of one specification allowed.
name	string	Name of the new role. Required when using createRole().
permissionDataObjs	permissionDataObj	Name of a permission to associate with the new role. See getAllPermissions(). Zero or more permissions allowed. When updating role permissions, respecify any permissions you want to retain.

### 3.5.81. Operation: updateSignInConfiguration

#### 3.5.81.1. Name

updateSignInConfiguration

#### 3.5.81.2. Description

Update sign-in settings. These configurations are identical to Sign In Settings (a set of System Configuration settings) in the Coverity Connect UI.

#### 3.5.81.3. Parameters

signInSettingsDataObj

**Type:** signInSettingsDataObj

Passes Coverity Connect sign-in settings.

Field name	Type	Description
allowPasswordRecovery	boolean	If set to true, users who are locked out due to incorrect password attempts can request their password through email. Requires previous email configuration.
authenticationMethod	string	Sets the sign-in authentication method. You can specify one of these strings: LDAP, KERBEROS, or REVERSE_PROXY.
disableLocalPasswordAuth	boolean	If set to true, disables local account access and uses LDAP for authentication. Requires previous LDAP configuration.
enableLdapAuth	boolean	If set to true, access for any user in LDAP (including Active Directory users) is allowed. To work, the setting

Field name	Type	Description
		requires an LDAP configuration. Coverity Connect uses local accounts by default.
ldapUserAutoCreate	boolean	If set to true, allows creation of users in Coverity Connect upon successful authentication with the LDAP server. Requires an LDAP configuration and that enableLdapAuth is also true.
limitFailedSignIns	boolean	If set to true, a specified number of failed name-password sign-in attempts (exceeding maxFailedSignInAttempts) will lock out a user. Once this happens, unless password recovery is enabled, the administrator must reset the password for this user.
maxFailedSignInAttempts	int	Number of failed name-password sign-in attempts that are allowed before locking out a user. See limitFailedSignins.
maxSessionIdleTime	int	Maximum period of inactivity allowed before the session times out. Default value is 120 mins.
requireLdapGroupMembership	boolean	If true, only allows the creation of LDAP users that are members of imported LDAP groups. This setting provides for backward compatibility with Microsoft Active Directory products that require LDAP users to be members of an LDAP group.

### 3.5.82. Operation: updateSnapshotInfo

#### 3.5.82.1. Name

updateSnapshotInfo

#### 3.5.82.2. Description

Update a snapshot.

#### 3.5.82.3. Parameters

snapshotId

**Type:** snapshotIdDataObj

Identifier for a snapshot.

Field name	Type	Description
id	long	Numeric identifier for the snapshot. Required.

snapshotData

**Type:** snapshotInfoDataObj

Specification for snapshot information.

Field name	Type	Description
analysisCommandLine	string	Command line used when running the analysis of the source code associated with the snapshot.
analysisConfiguration	string	Configuration file used when running the analysis of the source code associated with the snapshot.
analysisHost	string	Name of the host machine used to run the analysis of the source code associated with the snapshot.
analysisIntermediateDir	string	Intermediate directory that contained the analysis results committed in this snapshot.
analysisInternalVersion	string	Internal version of Coverity Analysis used to run the analysis.
analysisTime	long	Duration of the analysis in seconds.
analysisVersion	string	Licensed version of Coverity Analysis used to run the analysis.
buildCommandLine	string	Build command used to compile the source code for analysis.
buildConfiguration	string	Configuration file used when compiling source code for analysis.
buildHost	string	Host machine used to run the build.
buildIntermediateDir	string	Intermediate directory into which the build was emitted.
buildTime	long	Duration of the build.
codeVersionDate	dateTime	Date and time of the analyzed code version according to the source control system, or if that is not available, the date and time when the build was captured.
commitUser	string	Username of user who committed the analysis results to the database.

Field name	Type	Description
dateCreated	dateTime	Date and time that the snapshot was created in the database.
description	string	Description of the snapshot.
enabledCheckers	string	List of checkers that were used in the analysis.
hasSummaries	boolean	True if this snapshot contains interprocedural analysis summaries that can be used to accurately analyze subsets of the code in isolation.
impactHashVersion	int	Internal field.
portableAnalysisSettings	string	This field contains information about the analysis settings used to create the snapshot, and is used by Desktop Analysis to imitate those settings. It is only meant to be used by cov-run-desktop and its format is subject to change.
purgedOfDetails	boolean	Value of <i>true</i> if the snapshot details have been purged.
snapshotId	snapshotIdDataObj	Identifier for the snapshot.
sourceVersion	string	Version of the source code. Present only if passed when committing analysis results (passed by the <i>--version</i> option of cov-commit-defects command or through an advanced commit setting in Coverity Wizard).
target	string	Target platform of the source code (for example, i386). Present only if passed when committing analysis results (passed by the <i>--target</i> option of the cov-commit-defects command or through an advanced commit setting in Coverity Wizard).

### 3.5.83. Operation: updateStream

#### 3.5.83.1. Name

updateStream

#### 3.5.83.2. Description

Update a stream specification.

### 3.5.83.3. Parameters

streamId

**Type:** streamIdDataObj

Identifier for a stream.

Field name	Type	Description
name	string	Required. Name of the stream.

streamSpec

**Type:** streamSpecDataObj

Specification for a stream.

Field name	Type	Description
allowCommitWithoutPassword	boolean	Boolean that indicates whether Coverity Connect will accept new commits of analysis results (through cov-commit-defects) to the stream without requiring a password. When creating or updating a stream, check your company's security policy before setting to <i>true</i> . If set to <i>true</i> , any user can commit analysis results to the stream.
analysisVersionOverride	string	Specifies the updated Coverity Analysis version that may retrieve data from this stream.
autoDeleteOnExpiry	boolean	Set to <i>true</i> if the stream should be deleted after a period of inactivity. The default period is 28 days. (Note that the <i>stream.expiration.inactivity.days</i> period is configurable through the <i>cim.properties</i> file. The stream will not be deleted unless it contains at least one snapshot.) Defaults to <i>false</i> with <i>createStream()</i> and <i>createStreamInProject()</i> .
componentMapId	componentMapIdDataObj	Name of the component map with which to associate the stream. Defaults to the <i>Default Component Map</i> with <i>createStream()</i> and <i>createStreamInProject()</i> .
description	string	Description of the stream.

Field name	Type	Description
enableDesktopAnalysis	boolean	If true, this stream is able to accept analysis summary information during commit.
language	string	The programming language of the source code files associated with the new stream. Required with createStream() and createStreamInProject().
name	string	Name of the new stream. Required with createStream() and createStreamInProject().
outdated	boolean	If true, the filter applies to streams that have been designated as <i>outdated</i> and are hidden from the Coverity Connect UI. If false, the filter applies to non-outdated streams.
ownerAssignmentOption	string	Specifies the updated Coverity Desktop plug-in version that may retrieve data from this stream.
pluginVersionOverride	string	
roleAssignments	roleAssignmentDataObj	Role to associate with the new stream. By default, the username of the stream creator is assigned the <i>streamOwner</i> role for the new stream. See getAllRoles(), getRole(), and getAllPermissions(). If updating role assignments, respecify any that you want to retain.
summaryExpirationDays	int	Specifies the number of days analysis summaries remain in this stream before being automatically removed from their relative snapshot. If you do not want summaries to be deleted, set summaryExpirationDays to 0. This will reset the value to null, and analysis summaries will not expire.
trriageStoreId	trriageStoreIdDataObj	Identifier for the triage store to associate with the new stream. Required with createStream()and createStreamInProject(). Maximum of one triage store name specification allowed.

Field name	Type	Description
versionMismatchMessage	string	Provides additional instructions to analysis users that are using an incorrect version of Coverity Analysis or the Coverity Desktop plug-in.

### 3.5.84. Operation: updateTriageStore

#### 3.5.84.1. Name

updateTriageStore

#### 3.5.84.2. Description

Update a triage store specification.

#### 3.5.84.3. Parameters

trriageStoreId

**Type:** triageStoreIdDataObj

Identifier for a triage store.

Field name	Type	Description
name	string	Required. Name of the triage store.

trriageStoreSpec

**Type:** triageStoreSpecDataObj

Specification for a triage store.

Field name	Type	Description
description	string	Description of the triage store.
name	string	Name of the triage store. Required with createTriageStore().
roleAssignments	roleAssignmentDataObj	Role to associate with the triage store at the global level. See getAllRoles(), getRole(), and getAllPermissions(). Zero or more role associations are allowed. If updating role assignments, respecify any that you want to retain.

### 3.5.85. Operation: updateUser

#### 3.5.85.1. Name

updateUser

### 3.5.85.2. Description

Update a user specification.

### 3.5.85.3. Parameters

username

**Type:** string

userSpec

**Type:** userSpecDataObj

Field name	Type	Description
disabled	boolean	To prevent the user from logging in to the UI, set to <i>true</i> . Defaults to <i>false</i> for <code>createUser()</code> .
domain	serverDomainIdDataObj	For an LDAP user only, the LDAP domain of user.
email	string	Email address of the user.
familyName	string	Last name of the user.
givenName	string	First name of the user.
groupNames	groupIdDataObj	Name of an existing group to which the user should belong. Zero or more group associations allowed. <i>Supported when updating a user, not when creating a user.</i>
local	boolean	If an LDAP user, set to <i>false</i> . Defaults to <i>true</i> with <code>createUser()</code> , specifying a local (non-LDAP) user.
locale	string	The locale of the user. Defaults to <i>en-US</i> with <code>createUser()</code> .
locked	boolean	To lock out the new user, set to <i>true</i> . Unless password recovery is enabled through the UI, the administrator must reset the password for the locked out user before the user can log in. Defaults to <i>false</i> with <code>createUser()</code> .
password	string	Password for the user. Required with <code>createUser()</code> .
roleAssignments	roleAssignmentDataObj	Role to associate with the new user at the global level. See <code>getAllRoles()</code> , <code>getRole()</code> , and <code>getAllPermissions()</code> . Zero or more role associations are

Field name	Type	Description
username	string	<p>allowed. If updating role assignments, respecify any that you want to retain.</p> <p>Required. Username for/of the user. Any capitalized (upper case) letters in the name will be converted to lower case when using createUser(). Maximum of one name specification is allowed.</p>

---

## 3.6. Complex Types: Configuration Service

### 3.6.1. Description

Configuration Service data objects.

### 3.6.2. Complex Types

Name	Description
attributeDefinitionDataObj	Returns the properties of an attribute.
attributeDefinitionIdDataObj	Identifier for an attribute.
attributeDefinitionSpecDataObj	Specification used to set the properties of an attribute.
attributeValueChangeSpecDataObj	Specification of the values for a LIST_OF_VALUES (Pick list) attribute type.
attributeValueDataObj	Metadata that pertain to a value of a LIST_OF_VALUES (Pick list) attribute type.
attributeValueIdDataObj	Identifer for an attribute value.
attributeValueSpecDataObj	Specification that indicates whether an attribute value is deprecated.
backupConfigurationDataObj	Specification for Coverity Connect database backup settings.
commitStateDataObj	Returns data that indicates whether new commits of analysis data are allowed in a stream and how many commits are in progress.
componentDataObj	Component data.
componentDefectRuleDataObj	Rule that assigns an owner to the software issues found in a specified component.
componentIdDataObj	Identifier for a component.
componentMapDataObj	Returns component map data.
componentMapFilterSpecDataObj	Filter used to return a matching set of component maps.
componentMapIdDataObj	Identifier for a component map.
componentMapSpecDataObj	Specification for a component map.
componentPathRuleDataObj	Pattern used to match the path to one or more source code files associated with a specified component.

---

Name	Description
configurationDataObj	Returns system configuration data.
CovRemoteServiceException	Error data typically for invalid requests.
deleteSnapshotJobInfoDataObj	Returns the status of a snapshot deletion request.
featureUpdateTimeDataObj	Update information on a feature.
getOutputFileForSnapshot	Specifies a snapshot output file.
getOutputFileForSnapshotResponse	Returns a snapshot output file.
groupDataObj	User group data.
groupFilterSpecDataObj	Filter properties used to return a matching set of user groups.
groupIdDataObj	Identifier for a user group.
groupsPageDataObj	Returned page of group records that includes the total number of records.
groupSpecDataObj	Specification for a user group.
ldapConfigurationDataObj	Returns LDAP configuration data.
ldapConfigurationSpecDataObj	Specification for an LDAP configuration.
licenseDataObj	Specification for license data.
licenseSpecDataObj	Specification for license file data.
licenseStateDataObj	Returns Coverity Desktop license state.
localizedValueDataObj	Object containaing a display name and ID for various issue attributes.
loggingConfigurationDataObj	Specification of logging configurations.
outputFileDataObj	
pageSpecDataObj	Specification for the page of records to return.
permissionDataObj	Permission data.
projectDataObj	Returns project data.
projectFilterSpecDataObj	Filter properties used to return a matching set of projects.
projectIdDataObj	Identifier for a project.
projectSpecDataObj	Specification for a project.
roleAssignmentDataObj	Role assignment data.
roleDataObj	Returns role data.
roleIdDataObj	Identifier for a role.
roleSpecDataObj	Specification for a role.
serverDomainIdDataObj	Identifier for an LDAP server domain.
signInSettingsDataObj	Specification for Coverity Connect sign-in settings.
skeletonizationConfigurationDataObj	Specification for snapshot purge settings.
snapshotFilterSpecDataObj	Filter properties used to return a matching set of snapshots.

Name	Description
snapshotIdDataObj	Identifier for a snapshot.
snapshotInfoDataObj	Snapshot information.
snapshotPurgeDetailsObj	Deprecated specification for snapshot purge settings.
standardAttributeDataObj	Returns the properties of a standard attribute.
standardAttributeIdDataObj	Identifier for a standard attribute.
standardAttributeValueDataObj	Metadata that pertain to a standard attribute value.
standardAttributeValueIdDataObj	Identifier for a standard attribute value.
streamDataObj	Stream data.
streamFilterSpecDataObj	Filter properties used to return a matching set of streams.
streamIdDataObj	Identifier for a stream.
streamSpecDataObj	Specification of a stream.
triageStoreDataObj	Returns triage store properties.
triageStoreFilterSpecDataObj	Filter properties used to return a matching set of triage stores.
triageStoreIdDataObj	Identifier of a triage store.
triageStoreSpecDataObj	Specification of a triage store.
userDataObj	Returned properties of a user.
userFilterSpecDataObj	Filter properties used to return a matching set of user records.
usersPageDataObj	Returned page of user records and count of records in the page.
userSpecDataObj	Specification for a user.
versionDataObj	Returns Coverity Connect version data.

### 3.6.3. Complex Type: attributeDefinitionDataObj

#### 3.6.3.1. Description

Returns the properties of an attribute.

#### 3.6.3.2. Derived By

Restricting anyType

#### 3.6.3.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
attributeDefinitionId	attributeDefinitionIdDataObj	Name of an attribute.

Component	Type	Description
attributeType	string	Attribute type.
builtIn	boolean	Value of true for a built-in attribute; otherwise, <i>false</i> .
configurableValues	attributeValueDataObj	For a LIST_OF_VALUES attribute type only: The set of values available to the attribute.
defaultValue	string	For a LIST_OF_VALUES attribute type only: The default attribute value.
description	string	Description of the attribute.
displayDescription	string	Description of the attribute that appears in the UI.
displayName	string	Name of the attribute that appears in the UI.
showInTriage	boolean	Value of <i>true</i> if the attribute appears in the Triage pane; otherwise, <i>false</i> .

### 3.6.4. Complex Type: attributeDefinitionIdDataObj

#### 3.6.4.1. Description

Identifier for an attribute.

#### 3.6.4.2. Derived By

Restricting anyType

#### 3.6.4.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Required. Name of the attribute.

### 3.6.5. Complex Type: attributeDefinitionSpecDataObj

#### 3.6.5.1. Description

Specification used to set the properties of an attribute.

#### 3.6.5.2. Derived By

Restricting anyType

### 3.6.5.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
attributeName	string	Name for the attribute. Required when using createAttribute().
attributeType	string	The type of attribute. Required when using createAttribute().
attributeValueChangeSpec	attributeValueChangeSpecDataObj	For a LIST_OF_VALUES attribute type only: The set of values available to the attribute.
defaultValue	string	For a LIST_OF_VALUES attribute type only: The default attribute value.
description	string	Description of the attribute.
showInTriage	boolean	If <i>true</i> , makes the attribute available for use in the Triage pane of the UI.

### 3.6.6. Complex Type: attributeValueChangeSpecDataObj

#### 3.6.6.1. Description

Specification of the values for a LIST\_OF\_VALUES (Pick list) attribute type.

#### 3.6.6.2. Derived By

Restricting anyType

#### 3.6.6.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
attributeValueIds	attributeValueIdDataObj	Automatically generated set of IDs for attribute values.
attributeValues	attributeValueSpecDataObjSet	Set of values available to an attribute.

### 3.6.7. Complex Type: attributeValueDataObj

#### 3.6.7.1. Description

Metadata that pertain to a value of a LIST\_OF\_VALUES (Pick list) attribute type.

### 3.6.7.2. Derived By

Restricting anyType

### 3.6.7.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
attributeValueld	attributeValueldDataObj	Identifier for an attribute value.
deprecated	boolean	Value of <i>true</i> if the attribute value is deprecated. Otherwise, <i>false</i> .
displayName	string	Name of the attribute that appears in the UI.
issueKindList	string	Issue kind. Multiple issue kinds allowed.

### 3.6.8. Complex Type: attributeValueldDataObj

#### 3.6.8.1. Description

Identifer for an attribute value.

#### 3.6.8.2. Derived By

Restricting anyType

#### 3.6.8.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Name of the automatically generated ID for the attribute value. Do not specify when creating an attribute value.

### 3.6.9. Complex Type: attributeValueSpecDataObj

#### 3.6.9.1. Description

Specification that indicates whether an attribute value is deprecated.

### 3.6.9.2. Derived By

Restricting anyType

### 3.6.9.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
deprecated	boolean	Value of <i>true</i> if the specified attribute value is deprecated. Otherwise, <i>false</i> .
name	string	Name of the attribute value.

## 3.6.10. Complex Type: backupConfigurationDataObj

### 3.6.10.1. Description

Specification for Coverity Connect database backup settings.

### 3.6.10.2. Derived By

Restricting anyType

### 3.6.10.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
backupLocation	string	Full path to a backup directory on the disk where Coverity Connect is installed.
backupTime	string	Time of the backup, for example: 01:15
fridayEnabled	boolean	Friday backup if set to true. Defaults to false.
mondayEnabled	boolean	Monday backup if set to true. Defaults to false.
saturdayEnabled	boolean	Saturday backup if set to true. Defaults to false.
sundayEnabled	boolean	Sunday backup if set to true. Defaults to false.
thursdayEnabled	boolean	Thursday backup if set to true. Defaults to false.
tuesdayEnabled	boolean	Tuesday backup if set to true. Defaults to false.
wednesdayEnabled	boolean	Wednesday backup if set to true. Defaults to false.

Component	Type	Description
-----------	------	-------------

### 3.6.11. Complex Type: commitStateDataObj

#### 3.6.11.1. Description

Returns data that indicates whether new commits of analysis data are allowed in a stream and how many commits are in progress.

#### 3.6.11.2. Derived By

Restricting anyType

#### 3.6.11.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
currentCommitCount	int	Number of commits currently in progress.
isAcceptingNewCommits	boolean	Value of <i>true</i> if new commits will be accepted. Otherwise, <i>false</i> .

### 3.6.12. Complex Type: componentDataObj

#### 3.6.12.1. Description

Component data.

#### 3.6.12.2. Derived By

Restricting anyType

#### 3.6.12.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
roleAssignments	roleAssignmentDataObj	Set of roles assigned to a user or group for the specified component.
componentId	componentIdDataObj	Component to which the roles and/or subscribers apply.

Component	Type	Description
subscribers	string	Set of usernames that subscribe to component notifications for this component map. Depends on proper usage of the notify() operation.

### 3.6.13. Complex Type: componentDefectRuleDataObj

#### 3.6.13.1. Description

Rule that assigns an owner to the software issues found in a specified component.

#### 3.6.13.2. Derived By

Restricting anyType

#### 3.6.13.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
componentId	componentIdDataObj	Component to which the defect rule applies.
defaultOwner	string	Default owner of the component.

### 3.6.14. Complex Type: componentIdDataObj

#### 3.6.14.1. Description

Identifier for a component.

#### 3.6.14.2. Derived By

Restricting anyType

#### 3.6.14.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Required. Name of a component in the form <b>componentMapName.componentName</b> (for example, myComponentMap.myComponent).

Component	Type	Description
-----------	------	-------------

### 3.6.15. Complex Type: componentMapDataObj

#### 3.6.15.1. Description

Returns component map data.

#### 3.6.15.2. Derived By

Restricting anyType

#### 3.6.15.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
componentMapId	componentMapIdDataObj	Name of the component map.
componentPathRules	componentPathRuleDataObj	Path to set of files that are associated with a component of the component map. Multiple paths are possible.
components	componentDataObj	Component associated with a component map. Multiple component associations allowed.
defectRules	componentDefectRuleDataObj	Default owner of CIDs associated with the specified component.
description	string	Description of the component map.

### 3.6.16. Complex Type: componentMapFilterSpecDataObj

#### 3.6.16.1. Description

Filter used to return a matching set of component maps.

#### 3.6.16.2. Derived By

Restricting anyType

#### 3.6.16.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
namePattern	string	Glob pattern matching the name of one or more component maps.

### 3.6.17. Complex Type: componentMapIdDataObj

#### 3.6.17.1. Description

Identifier for a component map.

#### 3.6.17.2. Derived By

Restricting anyType

#### 3.6.17.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Required. Name of the component map to delete or update.

### 3.6.18. Complex Type: componentMapSpecDataObj

#### 3.6.18.1. Description

Specification for a component map.

#### 3.6.18.2. Derived By

Restricting anyType

#### 3.6.18.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
componentMapName	string	Required. Name of the component map.
componentPathRules	componentPathRuleDataObj	Pattern matching a path to set of files to associate with a component for the component map. Multiple component map rules allowed.

Component	Type	Description
components	componentDataObj	Specification of the component that you are associating with the componentmap. Multiple component associations allowed. When <i>updating</i> (not creating) components, you must specify the complete list of your components, including the "Other" component because any component in the component map that does not appear in the update list will be deleted. Alternatively, you can pass <b>null</b> to retain all existing components during an update.
defectRules	componentDefectRuleDataObj	Default owner of the specified component. Each component can have an owner.
description	string	Description of the component map.
forceDeleteComponents	boolean	When set to <code>true</code> , components that do not appear in the update list and are chosen to be included in hierarchies will be deleted and removed from the hierarchies. When set to <code>false</code> , components that do not appear in the update list and are chosen to be included in hierarchies will <i>not</i> be deleted. In the latter case, an error message is returned.

### 3.6.19. Complex Type: componentPathRuleDataObj

#### 3.6.19.1. Description

Pattern used to match the path to one or more source code files associated with a specified component.

#### 3.6.19.2. Derived By

Restricting anyType

#### 3.6.19.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
componentId	componentIdDataObj	Required. Name of the component to which the <i>pathPattern</i> applies. Takes the form <i>[componentMapName.componentName]</i> .
pathPattern	string	Required. Path to one or more associated source code files.

### 3.6.20. Complex Type: configurationDataObj

#### 3.6.20.1. Description

Returns system configuration data.

#### 3.6.20.2. Derived By

Restricting anyType

#### 3.6.20.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
commitPort	long	Port number that supports commits of analysis results.
dbDialect	string	Database dialect.
dbDriver	string	Database driver.
issueExportUrl	string	URL used to handle exported defect information.
maindbName	string	Name of the database.
maindbUrl	string	URL of the database.
maindbUser	string	Authorized user of the database.

### 3.6.21. Complex Type: CovRemoteServiceException

#### 3.6.21.1. Description

Error data typically for invalid requests.

#### 3.6.21.2. Derived By

Restricting anyType

#### 3.6.21.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
errorCode	int	Error code associated with the error message.

Component	Type	Description
message	string	Error message.

### 3.6.22. Complex Type: deleteSnapshotJobInfoDataObj

#### 3.6.22.1. Description

Returns the status of a snapshot deletion request.

#### 3.6.22.2. Derived By

Restricting anyType

#### 3.6.22.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
snapshotId	long	Identifier for the snapshot. Available though the UI.
status	deleteSnapshotJobStatus	Indication of whether the snapshot deletion process succeeded or failed.

#### 3.6.22.4. Remarks

See also, deleteSnapshot().

### 3.6.23. Complex Type: featureUpdateTimeDataObj

#### 3.6.23.1. Description

Update information on a feature.

#### 3.6.23.2. Derived By

Restricting anyType

#### 3.6.23.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
featureName	string	Feature that was updated.
lastUpdateDate	dateTime	Date and time the feature was last updated.

### 3.6.24. Complex Type: getOutputFileForSnapshot

#### 3.6.24.1. Description

Specifies a snapshot output file.

#### 3.6.24.2. Derived By

Restricting anyType

#### 3.6.24.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
snapshotId	snapshotIdDataObj	Identifies the snapshot whose output file you want to retrieve.
fileName	string	Identifies the output file to retrieve.

### 3.6.25. Complex Type: getOutputFileForSnapshotResponse

#### 3.6.25.1. Description

Returns a snapshot output file.

#### 3.6.25.2. Derived By

Restricting anyType

#### 3.6.25.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
return	outputFileDataObj	The output file from the snapshot that matches the specified file name, or null if none is found.

### 3.6.26. Complex Type: groupDataObj

#### 3.6.26.1. Description

User group data.

#### 3.6.26.2. Derived By

Restricting anyType

#### 3.6.26.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
roleAssignments	roleAssignmentDataObj	Role assigned to the group.
local	boolean	Value of <i>true</i> if the group is local; <i>false</i> if LDAP.
name	groupIdDataObj	Name of the local or LDAP user group.
syncEnabled	boolean	Value of <i>false</i> if the group is a local group; defaults to <i>false</i> .

### 3.6.27. Complex Type: groupFilterSpecDataObj

#### 3.6.27.1. Description

Filter properties used to return a matching set of user groups.

#### 3.6.27.2. Derived By

Restricting anyType

#### 3.6.27.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
ldap	boolean	Value of <i>true</i> for LDAP groups only; otherwise, <i>false</i> .
namePattern	string	Glob pattern matching the name of one or more groups.
projectIdDataObj	projectIdDataObj	Name of a project with which the group must have a role association.

Component	Type	Description
userList	string	User name of a user that must belong to the group. Multiple names allowed.

### 3.6.28. Complex Type: groupIdDataObj

#### 3.6.28.1. Description

Identifier for a user group.

#### 3.6.28.2. Derived By

Restricting anyType

#### 3.6.28.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
displayName	string	The name of a user group. To retrieve an LDAP group, you use <groupname>@<LDAPserver>.
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group.
name	string	Required. Name of the local or LDAP group.

### 3.6.29. Complex Type: groupsPageDataObj

#### 3.6.29.1. Description

Returned page of group records that includes the total number of records.

#### 3.6.29.2. Derived By

Restricting anyType

#### 3.6.29.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
groups	groupDataObj	List of user groups returned by the request.

Component	Type	Description
totalNumberOfRecords	int	Total number of group records returned.

### 3.6.30. Complex Type: groupSpecDataObj

#### 3.6.30.1. Description

Specification for a user group.

#### 3.6.30.2. Derived By

Restricting anyType

#### 3.6.30.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
domain	serverDomainIdDataObj	For an LDAP user group only, the LDAP domain of group. Required when using createGroup() for an LDAP group. Maximum of one <i>domain</i> specification allowed.
local	boolean	Defaults to <i>true</i> , specifying a local (non-LDAP) group. If an LDAP group, set to <i>false</i> when using createGroup(). Maximum of one <i>local</i> specification allowed.
name	string	Name for the user group. Required when using createGroup().
roleAssignments	roleAssignmentDataObj	Role to associate with the group at the global level. See getAllRoles(), getRole(), and getAllPermissions(). Zero or more role associations allowed. If updating role assignments, respecify any global type roles that you want to retain.
syncEnabled	boolean	If a local group, set to <i>false</i> when using createRole(). Defaults to <i>true</i> . Applies to an LDAP group only.

### 3.6.31. Complex Type: ldapConfigurationDataObj

#### 3.6.31.1. Description

Returns LDAP configuration data.

### 3.6.31.2. Derived By

Restricting anyType

### 3.6.31.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
anonymousBind	boolean	Value of <i>true</i> if using an anonymous bind; otherwise, <i>false</i> to indicate a general bind with a shared bind DN.
baseDN	string	LDAP domain name in host format, such as <domain>.com or <corp.domain>.<com>, or base DN such as <i>ou=corp</i> .
bindName	string	The username needed to access the LDAP server.  Can be specified in <i>username@domain</i> or LDAP DN format.
bindPassword	string	Bind user password for LDAP user queries. Required for authenticated binding requests. If there is no password, and if your LDAP server is configured to accept unauthenticated binding requests, Coverity Connect will attempt to gain unauthenticated access to the LDAP server. If an unauthenticated LDAP connection fails, Coverity Connect will display an LDAP Server Configuration failed message and an explanation of the failure. You can find more information about the failure in the <i>cim.log</i> file.
groupFilter	string	Optional filter used when importing LDAP groups. Takes a valid RFC format. Example: <i>cn=eng*</i>
groupFullName	boolean	Members stored by full DN or Username.
groupMember	string	If <i>false</i> , group members are stored by <i>userName</i> . If <i>true</i> , members are stored by their full DN. Defaults to <i>true</i> .
groupName	string	Name attribute for the group. For Example: <i>cn</i>
groupObjectClass	string	LDAP <i>objectClass</i> value that identifies user groups. For OpenLDAP, the default is <i>groupofnames</i> . For Active Directory, the default is <i>group</i> . This field defines a component of the

Component	Type	Description
		LDAP user group search query that Coverity Connect creates.
groupSearchBase	string	The group search base DN to prepend to the base DN. Used for group searches. The base DN is configured as <i>Domain</i> as part of basic LDAP settings. For example: <i>cn=groups, ou=corp</i>
primary	boolean	Indicates whether this LDAP configuration is used. Note that multiple LDAP configurations are possible.
secureConnection	boolean	Value of <i>true</i> if using a secure connection. SSL or TLS protocol required.
serverDomain	string	The host name or host IP of the LDAP server. The name must be resolvable from the Coverity Connect host.
serverDomainIdDataObj	serverDomainIdDataObj	Identifier for the LDAP server domain.
serverPort	long	The TCP port number where the LDAP server listens for connections (default is 389).
tlsEnabled	boolean	Value of <i>true</i> if TLS security protocol is used.
userEmail	string	LDAP attribute that maps to the email address of the user.
userFirstName	string	LDAP attribute that maps to the first name (given name) of the user.
userLastName	string	LDAP attribute that maps to the last name (surname) of the user.
userName	string	Username for testing user search settings. Maps to the username in the database.
userObjectClass	string	LDAP <i>objectClass</i> associated with users on the LDAP server. For OpenLDAP, the default is <i>inetOrgPerson</i> . For Active Directory, the default is <i>user</i> .
userSearchBase	string	Relative DN to prepend to base DN to limit a user search. For example, <i>ou=Users</i> will search in <i>ou=Users, dc=ad, dc=coverity, dc=com</i> .

### 3.6.32. Complex Type: ldapConfigurationSpecDataObj

#### 3.6.32.1. Description

Specification for an LDAP configuration.

#### 3.6.32.2. Derived By

Restricting anyType

### 3.6.32.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
anonymousBind	boolean	Value of <i>true</i> if using an anonymous bind; otherwise, <i>false</i> to indicate a general bind with a shared bind DN.
baseDN	string	LDAP domain name in host format, such as <domain>.com or <corp.domain>.<com>, or base DN such as <i>ou=corp</i> .
bindName	string	The username needed to access the LDAP server.  You can specify the user in username@domain or LDAP DN format.
bindPassword	string	Bind user password for LDAP user queries. Required for authenticated binding requests. If you do not enter a password, and if your LDAP server is configured to accept unauthenticated binding requests, Coverity Connect will attempt to gain unauthenticated access to the LDAP server. If an unauthenticated LDAP connection fails, Coverity Connect will display an LDAP Server Configuration failed message and an explanation of the failure. You can find more information about the failure in the <i>cim.log</i> file.
displayName	string	Name that identifies the LDAP server in a multi-server setup. Appears in the UI.
groupFilter	string	Optional filter to use when importing LDAP groups. Takes a valid RFC format. Example: <i>cn=eng*</i>
groupFullName	boolean	If <i>false</i> , group members are stored by userName. If <i>true</i> , members are stored by their full DN. Defaults to <i>true</i> .
groupMember	string	LDAP attribute that defines the members of a group. Group members can be referred to by their DN or username.
groupName	string	Name attribute for the group. For Example: <i>cn</i>
groupObjectClass	string	LDAP <i>objectClass</i> value that identifies user groups. For OpenLDAP, the default is <i>groupofnames</i> . For Active Directory, the default

Component	Type	Description
		is <i>group</i> . This field defines a component of the LDAP user group search query that Coverity Connect creates.
groupSearchBase	string	The group search base DN to prepend to the base DN. Used for group searches. The base DN is configured as <i>Domain</i> as part of basic LDAP settings. For example: <i>cn=groups, ou=corp</i>
primary	boolean	Value of <i>true</i> if this configuration is the primary one. Defaults to <i>false</i> .
secureConnection	boolean	Value of <i>true</i> if using a secure connection. SSL or TLS protocol required. For information about LDAP server requirements, secure connections, importing LDAP users and groups, and other details about LDAP, refer to <i>Coverity Connect Usage and Administration Guide</i> .
serverDomain	string	The host name or host IP of the LDAP server. The name must be resolvable from the Coverity Connect host.
serverPort	long	The TCP port number where the LDAP server listens for connections.
tlsEnabled	boolean	Value of <i>true</i> if TLS security protocol is used.
userEmail	string	LDAP attribute that maps to the email address of the user.
userFirstName	string	LDAP attribute that maps to the first name (given name) of the user.
userLastName	string	LDAP attribute that maps to the last name (surname) of the user.
userName	string	Username for testing user search settings. Maps to the username in the database.
userObjectClass	string	LDAP <i>objectClass</i> associated with users on the LDAP server. For OpenLDAP, the default is <i>inetOrgPerson</i> . For Active Directory, the default is <i>user</i> .
userSearchBase	string	Relative DN to prepend to base DN to limit a user search. For example, <i>ou=Users</i> will search in <i>ou=Users, dc=ad, dc=coverity, dc=com</i> .

### 3.6.33. Complex Type: licenseDataObj

#### 3.6.33.1. Description

Specification for license data.

### 3.6.33.2. Derived By

Restricting anyType

### 3.6.33.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
customer	string	Customer name.
expirationDate	dateTime	Expiration date.
licenseEditionName	string	License type. Coverity component availability varies by license edition.
loc	long	Lines of code analyzed.
locLimit	long	Maximum lines of code allowed by the license.
userCount	int	Current number of users added to Coverity Connect.
userLimit	string	Maximum number of users allowed.

### 3.6.34. Complex Type: licenseSpecDataObj

#### 3.6.34.1. Description

Specification for license file data.

#### 3.6.34.2. Derived By

Restricting anyType

#### 3.6.34.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
licenseDataFile	string	Contents of the license file. See the example for details.

### 3.6.35. Complex Type: licenseStateDataObj

#### 3.6.35.1. Description

Returns Coverity Desktop license state.

### 3.6.35.2. Derived By

Restricting anyType

### 3.6.35.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
desktopAnalysisEnabled	boolean	Value of <i>true</i> if Coverity Desktop analysis is enabled; <i>false</i> , if not.

### 3.6.36. Complex Type: localizedValueDataObj

#### 3.6.36.1. Description

Object containaing a display name and ID for various issue attributes.

#### 3.6.36.2. Derived By

Restricting anyType

#### 3.6.36.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
displayName	string	Localized value suitable for displaying to the user.
name	string	ID value suitable for passing as a filter.

### 3.6.37. Complex Type: loggingConfigurationDataObj

#### 3.6.37.1. Description

Specification of logging configurations.

#### 3.6.37.2. Derived By

Restricting anyType

#### 3.6.37.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
accessControlLogging	boolean	Access control logging.
backgroundLogging	boolean	Background tasks logging.
bugTrackingSystemLogging	boolean	Bug tracking system logging.
commitLogging	boolean	Commit logging.
configurationLogging	boolean	Configuration logging.
databaseLogging	boolean	Database logging.
frameworkLogging	boolean	Framework logging.
internalLogging	boolean	Internal logging.
kerberosLogging	boolean	Kerberos logging.
metricsAndHistoryLogging	boolean	Metrics and history logging.
notificationLogging	boolean	Notifications logging.
performanceLogging	boolean	Performance logging.
policyManagerLogging	boolean	Policy Manager logging.
remoteConfigLogging	boolean	Remote Configuration logging.
requestPerformanceLogging	boolean	Request Performance logging.
skeletonizationLogging	boolean	Snapshot Details Purge logging.
testAdvisorLogging	boolean	Test Advisor logging.
triageLogging	boolean	Triage logging.
triageSynchLogging	boolean	Triage & CID Synchronization logging.
webLogging	boolean	Web logging.
webServicesLogging	boolean	Web Services logging.

### 3.6.38. Complex Type: outputFileDataObj

#### 3.6.38.1. Derived By

Restricting anyType

#### 3.6.38.2. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
contents	base64Binary	File contents.
name	string	File name.

### 3.6.39. Complex Type: pageSpecDataObj

#### 3.6.39.1. Description

Specification for the page of records to return.

#### 3.6.39.2. Derived By

Restricting anyType

#### 3.6.39.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
pageSize	int	Required. Number of records to return. Maximum of 1000.
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Do not specify with getGroups().  Name of the field to use for sorting results. For example, the dateCreated or familyName field for user records returned by getUsers(). You cannot sort by a field that can appear multiple times for a single record (for example, a single user record can have multiple group fields).  For examples of sortField values, see the remark for this data object.
startIndex	int	Zero-based index of user records to return. Defaults to 0. You might use this field if there are more than 1000 users. You could make separate calls that start the index at 1000 or 2000, for example, until you capture separate lists of all of the users.

#### 3.6.39.4. Remarks

A complete list of sortList values is not available at this time. In general, the following sort field values are valid when getting merged defects: action, classification, severity, status, id, Fix Target, displayFunction, Legacy, displayFile, component, lastDetected, lastTriaged, lastFixed, firstDetected, owner, Ext. Reference, checker. In addition, custom attribute names are valid. The filters available are generally limited to the attributes available in the query. So snapshot scope does not have a lastDetected field, but project scope does.

### 3.6.40. Complex Type: permissionDataObj

#### 3.6.40.1. Description

Permission data.

#### 3.6.40.2. Derived By

Restricting anyType

#### 3.6.40.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
permissionValue	string	A permission value.

### 3.6.41. Complex Type: projectDataObj

#### 3.6.41.1. Description

Returns project data.

#### 3.6.41.2. Derived By

Restricting anyType

#### 3.6.41.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
streams	streamDataObj	Name of a stream that is associated with the project.
streamLinks	streamDataObj	Name of a stream link that is associated with the project.
roleAssignments	roleAssignmentDataObj	Role of a user or group that is associated with the project.
dateCreated	dateTime	Date and time that the project was created in the database.
dateModified	dateTime	Date and time that the project was last modified. Can be the creation date and time.
description	string	Description of the project.

Component	Type	Description
id	projectIdDataObj	Identifier for the project.
projectKey	long	Unique numeric identifier for the project.
userCreated	string	Name of the user who created the project.
userModified	string	Name of the user who last updated the project.

### 3.6.42. Complex Type: projectFilterSpecDataObj

#### 3.6.42.1. Description

Filter properties used to return a matching set of projects.

#### 3.6.42.2. Derived By

Restricting anyType

#### 3.6.42.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
descriptionPattern	string	Glob pattern matching the description of one or more projects.
includeChildren	boolean	Value of <i>false</i> if the results <i>should not</i> include roles and other properties associated with the project. Defaults to <i>true</i> .
includeStreams	boolean	Value of <i>false</i> if the results <i>should not</i> include streams associated with the project. Defaults to <i>true</i> .
namePattern	string	Glob pattern matching the name of one or more projects.

### 3.6.43. Complex Type: projectIdDataObj

#### 3.6.43.1. Description

Identifier for a project.

#### 3.6.43.2. Derived By

Restricting anyType

#### 3.6.43.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Required. Name of the project.

### 3.6.44. Complex Type: projectSpecDataObj

#### 3.6.44.1. Description

Specification for a project.

#### 3.6.44.2. Derived By

Restricting anyType

#### 3.6.44.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
streams	streamIdDataObj	Name of an existing stream to associate with the project. When updating a project, you must list all of the streams that you want to retain except for streams with the __internal prefix to their name, such as Dynamic Analysis streams. You might see such internal streams listed when you call getProjects(). Zero or more stream name associations allowed. See the remark for additional details.
streamLinks	streamIdDataObj	Name of a new or existing stream link to associate with the project. (Note that a stream link and the stream to which it links must belong to separate projects.) Zero or more stream link name associations allowed.
description	string	Description of the project.
name	string	Name for the project. Required when using createProject().
roleAssignments	roleAssignmentDataObj	Role to associate with the project. See getAllRoles(), getRole(), and getAllPermissions(). By default, the username of the project creator is assigned the <i>projectOwner</i> role for the new stream. If updating role assignments, respecify any that you want to retain. Zero or more role associations allowed.

Component	Type	Description
-----------	------	-------------

### 3.6.45. Complex Type: roleAssignmentDataObj

#### 3.6.45.1. Description

Role assignment data.

#### 3.6.45.2. Derived By

Restricting anyType

#### 3.6.45.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
groupId	groupIdDataObj	Identifier for the user group. Used only if the role applies to a group, not a user.
roleAssignmentType	string	Role assignment type.
roleId	roleIdDataObj	Identifier for a role.
type	string	Role type.
username	string	User name associated with <i>user</i> roleId only. Does not apply to <i>group</i> roleId.

### 3.6.46. Complex Type: roleDataObj

#### 3.6.46.1. Description

Returns role data.

#### 3.6.46.2. Derived By

Restricting anyType

#### 3.6.46.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
deletable	boolean	Value of <i>true</i> if a role can be deleted; otherwise, <i>false</i> .

Component	Type	Description
description	string	Description of the role.
editable	boolean	Value of <i>true</i> if a role can be edited; otherwise, <i>false</i> .
permissionDataObjs	permissionDataObj	List of permissions associated with a role.
roleId	roleIdDataObj	Identifier for a role.

### 3.6.47. Complex Type: roleIdDataObj

#### 3.6.47.1. Description

Identifier for a role.

#### 3.6.47.2. Derived By

Restricting anyType

#### 3.6.47.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Required. Name of the role.

### 3.6.48. Complex Type: roleSpecDataObj

#### 3.6.48.1. Description

Specification for a role.

#### 3.6.48.2. Derived By

Restricting anyType

#### 3.6.48.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
deletable	boolean	If the new role can be deleted, set to <i>true</i> . If not, set to <i>false</i> . Required when using createRole(). Maximum of one specification allowed.
description	string	Description of the role.

Component	Type	Description
editable	boolean	Value of <i>false</i> not allowed. Requires value of <i>true</i> when using createRole(). Maximum of one specification allowed.
name	string	Name of the new role. Required when using createRole().
permissionDataObjs	permissionDataObj	Name of a permission to associate with the new role. See getAllPermissions(). Zero or more permissions allowed. When updating role permissions, respecify any permissions you want to retain.

### 3.6.49. Complex Type: serverDomainIdDataObj

#### 3.6.49.1. Description

Identifier for an LDAP server domain.

#### 3.6.49.2. Derived By

Restricting anyType

#### 3.6.49.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Name of the LDAP server domain.

### 3.6.50. Complex Type: signInSettingsDataObj

#### 3.6.50.1. Description

Specification for Coverity Connect sign-in settings.

#### 3.6.50.2. Derived By

Restricting anyType

#### 3.6.50.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
allowPasswordRecovery	boolean	If set to true, users who are locked out due to incorrect password attempts can request their password through email. Requires previous email configuration.
authenticationMethod	string	Sets the sign-in authentication method. You can specify one of these strings: LDAP, KERBEROS, or REVERSE_PROXY.
disableLocalPasswordAuth	boolean	If set to true, disables local account access and uses LDAP for authentication. Requires previous LDAP configuration.
enableLdapAuth	boolean	If set to true, access for any user in LDAP (including Active Directory users) is allowed. To work, the setting requires an LDAP configuration. Coverity Connect uses local accounts by default.
ldapUserAutoCreate	boolean	If set to true, allows creation of users in Coverity Connect upon successful authentication with the LDAP server. Requires an LDAP configuration and that enableLdapAuth is also true.
limitFailedSignIns	boolean	If set to true, a specified number of failed name-password sign-in attempts (exceeding maxFailedSignInAttempts) will lock out a user. Once this happens, unless password recovery is enabled, the administrator must reset the password for this user.
maxFailedSignInAttempts	int	Number of failed name-password sign-in attempts that are allowed before locking out a user. See limitFailedSignins.
maxSessionIdleTime	int	Maximum period of inactivity allowed before the session times out. Default value is 120 mins.
requireLdapGroupMemberships	boolean	If true, only allows the creation of LDAP users that are members of imported LDAP groups. This setting provides for backward compatibility with Microsoft Active Directory products that require LDAP users to be members of an LDAP group.

### 3.6.51. Complex Type: skeletonizationConfigurationDataObj

#### 3.6.51.1. Description

Specification for snapshot purge settings.

#### 3.6.51.2. Derived By

Restricting anyType

### 3.6.51.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
daysBeforeSkeletonization	int	Age (in number of days) at which the details of a snapshot can be purged.
fridayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
minSnapshotsToKeep	int	Number of snapshots that must retain their details. Minimum of one is required.
mondayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
saturdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
sundayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
thursdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
time	string	Time of day at which the purge should take place. Example for 5:00 a.m.: <i>05:00</i>
tuesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
wednesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.

### 3.6.52. Complex Type: snapshotFilterSpecDataObj

#### 3.6.52.1. Description

Filter properties used to return a matching set of snapshots.

#### 3.6.52.2. Derived By

Restricting anyType

#### 3.6.52.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
descriptionPattern	string	Glob pattern matching the description of one or more snapshots.
endDate	dateTime	Date (and optionally, time) on or before which the snapshot was created. Serves as an upper bound on the IDs to return. (If you do not enter a time, the system is likely to assume 12:00 a.m.) See the sample request below.
hasSummaries	boolean	If present, only snapshots with a hasSummaries attribute that is equal to the specified value will be returned. If absent, no filtering on hasSummaries takes place.
lastBeforeCodeVersionDate	dateTime	If present, only one snapshot will be returned, specifically, the snapshot with the latest codeVersionDate among those that are before or equal to the specified date. If there is no such snapshot, then the call will return an empty set.
startDate	dateTime	Date (and optionally, time) on or after when the snapshot was created. Serves as a lower bound on the IDs to return. (If you do not enter a time, the system is likely to assume 12:00 a.m.) See the sample request below.
targetPattern	string	Glob pattern matching the target of the snapshot.
versionPattern	string	Glob pattern matching the version of the snapshot.

### 3.6.53. Complex Type: snapshotIdDataObj

#### 3.6.53.1. Description

Identifier for a snapshot.

#### 3.6.53.2. Derived By

Restricting anyType

#### 3.6.53.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
id	long	Numeric identifier for the snapshot. Required.

Component	Type	Description
-----------	------	-------------

### 3.6.54. Complex Type: snapshotInfoDataObj

#### 3.6.54.1. Description

Snapshot information.

#### 3.6.54.2. Derived By

Restricting anyType

#### 3.6.54.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
analysisCommandLine	string	Command line used when running the analysis of the source code associated with the snapshot.
analysisConfiguration	string	Configuration file used when running the analysis of the source code associated with the snapshot.
analysisHost	string	Name of the host machine used to run the analysis of the source code associated with the snapshot.
analysisIntermediateDir	string	Intermediate directory that contained the analysis results committed in this snapshot.
analysisInternalVersion	string	Internal version of Coverity Analysis used to run the analysis.
analysisTime	long	Duration of the analysis in seconds.
analysisVersion	string	Licensed version of Coverity Analysis used to run the analysis.
buildCommandLine	string	Build command used to compile the source code for analysis.
buildConfiguration	string	Configuration file used when compiling source code for analysis.
buildFailureCount	int	Number of build failures
buildHost	string	Host machine used to run the build.
buildIntermediateDir	string	Intermediate directory into which the build was emitted.
buildSuccessCount	int	Number of build successes
buildTime	long	Duration of the build.

Component	Type	Description
codeVersionDate	dateTime	Date and time of the analyzed code version according to the source control system, or if that is not available, the date and time when the build was captured.
commitUser	string	Username of user who committed the analysis results to the database.
dateCreated	dateTime	Date and time that the snapshot was created in the database.
description	string	Description of the snapshot.
enabledCheckers	string	List of checkers that were used in the analysis.
hasSummaries	boolean	True if this snapshot contains interprocedural analysis summaries that can be used to accurately analyze subsets of the code in isolation.
impactHashVersion	int	Internal field.
portableAnalysisSettings	string	This field contains information about the analysis settings used to create the snapshot, and is used by Desktop Analysis to imitate those settings. It is only meant to be used by cov-run-desktop and its format is subject to change.
purgedOfDetails	boolean	Value of <i>true</i> if the snapshot details have been purged.
snapshotId	snapshotIdDataObj	Identifier for the snapshot.
sourceVersion	string	Version of the source code. Present only if passed when committing analysis results (passed by the <i>--version</i> option of cov-commit-defects command or through an advanced commit setting in Coverity Wizard).
target	string	Target platform of the source code (for example, i386). Present only if passed when committing analysis results (passed by the <i>--target</i> option of the cov-commit-defects command or through an advanced commit setting in Coverity Wizard).

### 3.6.55. Complex Type: snapshotPurgeDetailsObj

#### 3.6.55.1. Description

Deprecated specification for snapshot purge settings.

#### 3.6.55.2. Derived By

Restricting anyType

### 3.6.55.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
daysBeforeSkeletonization	int	Age (in number of days) at which the details of a snapshot can be purged. Defaults to 0.
fridayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
minSnapshotsToKeep	int	Number of snapshots that must retain their details. Defaults to 0.
mondayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
saturdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
sundayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
thursdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
timeOfDay	string	Require Time of day at which the purge should take place. Example for 5:00 a.m.: <i>05:00</i>
tuesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.
wednesdayEnabled	boolean	Value of <i>true</i> if a purge should take place on this day. Value of <i>false</i> if not.

### 3.6.56. Complex Type: standardAttributeDataObj

#### 3.6.56.1. Description

Returns the properties of a standard attribute.

#### 3.6.56.2. Derived By

Restricting anyType

#### 3.6.56.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
standardAttributeIdDataObj	standardAttributeIdDataObj	Name of the standard attribute.
standardAttributeValues	standardAttributeValueDataObj	set of values available to the standard attribute.

### 3.6.57. Complex Type: standardAttributeIdDataObj

#### 3.6.57.1. Description

Identifier for a standard attribute.

#### 3.6.57.2. Derived By

Restricting anyType

#### 3.6.57.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Name of the standard attribute.

### 3.6.58. Complex Type: standardAttributeValueDataObj

#### 3.6.58.1. Description

Metadata that pertain to a standard attribute value.

#### 3.6.58.2. Derived By

Restricting anyType

#### 3.6.58.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
description	string	Description of a standard attribute value.

### 3.6.59. Complex Type: standardAttributeValuelIdDataObj

#### 3.6.59.1. Description

Identifier for a standard attribute value.

### 3.6.59.2. Derived By

Restricting anyType

### 3.6.59.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Name of the standard attribute value.

### 3.6.60. Complex Type: streamDataObj

#### 3.6.60.1. Description

Stream data.

#### 3.6.60.2. Derived By

Restricting anyType

#### 3.6.60.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
autoDeleteOnExpiry	boolean	A value of <i>true</i> if the stream should be deleted after a period of inactivity; otherwise, <i>false</i> . The default inactivity period is 28 days. (Note that the <i>stream.expiration.inactivity.days</i> period is configurable through the <i>cim.properties</i> file. The stream will not be deleted unless it contains at least one snapshot.)
componentMapId	componentMapIdDataObj	Name of the component map with which the stream is associated. Same as the map for the copied stream.
description	string	Description of the stream. Same as description of the copied stream.
id	streamIdDataObj	Identifier for the stream. Automatically generated when you use <code>copyStream(): [copied_stream_name] copy [#]</code>

Component	Type	Description
language	string	Programming language of the stream. Same as the language of the copied stream.
outdated	boolean	If true, the stream has been designated as <i>outdated</i> and is hidden from the Coverity Connect UI.
primaryProjectId	projectIdDataObj	Name of the project with which the stream is associated. The copy is automatically associated with the same primary project as the source stream. This field is not returned when copying a stream.
trriageStoreId	trriageStoreIdDataObj	Identifier for the triage store with which the stream is associated.
roleAssignments	roleAssignmentDataObj	Set of roles associated with the new stream. The copy is assigned the same roles that are assigned to the source stream. This field is not returned when copying a stream.
analysisVersionOverride	string	If necessary, specifies the updated Coverity Analysis version that may retrieve data from this stream.
summaryExpirationDays	int	Specifies the number of days analysis summaries remain in this stream before being automatically removed from their relative snapshots. If null, summaries do not expire, except in cases of snapshot deletion or skeletonization.
pluginVersionOverride	string	If necessary, specifies the updated Coverity Desktop plug-in version that may retrieve data from this stream.
versionMismatchMessage	string	Provides additional instructions to analysis users that are using an incorrect version of Coverity Analysis or the Coverity Desktop plug-in.
enableDesktopAnalysis	boolean	If true, this stream is able to accept analysis summaries during commit.
ownerAssignmentOption	string	

### 3.6.61. Complex Type: streamFilterSpecDataObj

#### 3.6.61.1. Description

Filter properties used to return a matching set of streams.

#### 3.6.61.2. Derived By

Restricting anyType

### 3.6.61.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
languageList	string	Programming language matching that of one or more streams. Zero or more language filters allowed.
descriptionPattern	string	Glob pattern matching the description of one or more streams.
namePattern	string	Glob pattern matching the name of one or more streams.

### 3.6.62. Complex Type: streamIdDataObj

#### 3.6.62.1. Description

Identifier for a stream.

#### 3.6.62.2. Derived By

Restricting anyType

#### 3.6.62.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Required. Name of the stream.

### 3.6.63. Complex Type: streamSpecDataObj

#### 3.6.63.1. Description

Specification of a stream.

#### 3.6.63.2. Derived By

Restricting anyType

### 3.6.63.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
allowCommitWithoutPassword	boolean	Boolean that indicates whether Coverity Connect will accept new commits of analysis results (through cov-commit-defects) to the stream without requiring a password. When creating or updating a stream, check your company's security policy before setting to <i>true</i> . If set to <i>true</i> , any user can commit analysis results to the stream.
analysisVersionOverride	string	Specifies the updated Coverity Analysis version that may retrieve data from this stream.
autoDeleteOnExpiry	boolean	Set to <i>true</i> if the stream should be deleted after a period of inactivity. The default period is 28 days. (Note that the <i>stream.expiration.inactivity.days</i> period is configurable through the <i>cim.properties</i> file. The stream will not be deleted unless it contains at least one snapshot.) Defaults to <i>false</i> with <i>createStream()</i> and <i>createStreamInProject()</i> .
componentMapId	componentMapIdDataObj	Name of the component map with which to associate the stream. Defaults to the <i>Default Component Map</i> with <i>createStream()</i> and <i>createStreamInProject()</i> .
description	string	Description of the stream.
enableDesktopAnalysis	boolean	If true, this stream is able to accept analysis summary information during commit.
language	string	The programming language of the source code files associated with the new stream. Required with <i>createStream()</i> and <i>createStreamInProject()</i> .
name	string	Name of the new stream. Required with <i>createStream()</i> and <i>createStreamInProject()</i> .
outdated	boolean	If true, the filter applies to streams that have been designated as <i>outdated</i> and are hidden from the Coverity Connect UI. If false, the filter applies to non-outdated streams.
ownerAssignmentOption	string	
pluginVersionOverride	string	Specifies the updated Coverity Desktop plug-in version that may retrieve data from this stream.

Component	Type	Description
roleAssignments	roleAssignmentDataObj	Role to associate with the new stream. By default, the username of the stream creator is assigned the <i>streamOwner</i> role for the new stream. See <code>getAllRoles()</code> , <code>getRole()</code> , and <code>getAllPermissions()</code> . If updating role assignments, specify any that you want to retain.
summaryExpirationDays	int	Specifies the number of days analysis summaries remain in this stream before being automatically removed from their relative snapshot. If you do not want summaries to be deleted, set <code>summaryExpirationDays</code> to 0. This will reset the value to null, and analysis summaries will not expire.
triageStoreId	triageStoreIdDataObj	Identifier for the triage store to associate with the new stream. Required with <code>createStream()</code> and <code>createStreamInProject()</code> . Maximum of one triage store name specification allowed.
versionMismatchMessage	string	Provides additional instructions to analysis users that are using an incorrect version of Coverity Analysis or the Coverity Desktop plug-in.

### 3.6.64. Complex Type: triageStoreDataObj

#### 3.6.64.1. Description

Returns triage store properties.

#### 3.6.64.2. Derived By

Restricting anyType

#### 3.6.64.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
roleAssignments	roleAssignmentDataObj	Set of roles assigned to the retrieved triage store.
description	string	Description of the triage store.
id	triageStoreIdDataObj	Name of a retrieved triage store.
streamIds	streamIdDataObj	Identifier for a stream associated with the specified triage store.

### 3.6.65. Complex Type: triageStoreFilterSpecDataObj

#### 3.6.65.1. Description

Filter properties used to return a matching set of triage stores.

#### 3.6.65.2. Derived By

Restricting anyType

#### 3.6.65.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
descriptionPattern	string	Glob pattern matching the description of one or more triage stores.
namePattern	string	Glob pattern matching one or more names of the set of triage stores to retrieve.

### 3.6.66. Complex Type: triageStoreIdDataObj

#### 3.6.66.1. Description

Identifier of a triage store.

#### 3.6.66.2. Derived By

Restricting anyType

#### 3.6.66.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Required. Name of the triage store.

### 3.6.67. Complex Type: triageStoreSpecDataObj

#### 3.6.67.1. Description

Specification of a triage store.

### 3.6.67.2. Derived By

Restricting anyType

### 3.6.67.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
description	string	Description of the triage store.
name	string	Name of the triage store. Required with createTriageStore().
roleAssignments	roleAssignmentDataObj	Role to associate with the triage store at the global level. See getAllRoles(), getRole(), and getAllPermissions(). Zero or more role associations are allowed. If updating role assignments, respecify any that you want to retain.

### 3.6.68. Complex Type: userDataObj

#### 3.6.68.1. Description

Returned properties of a user.

#### 3.6.68.2. Derived By

Restricting anyType

#### 3.6.68.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
dateCreated	dateTime	Date and time that the user was created in the database.
dateModified	dateTime	Date and time that the user was modified. Can be the creation date and time.
disabled	boolean	Value of <i>true</i> if the user is disabled; <i>false</i> if not.
domain	serverDomainIdDataObj	If an LDAP user, the LDAP domain.

Component	Type	Description
email	string	Email address of the user.
familyName	string	Last name of the user.
givenName	string	First name of the user.
groups	string	List of groups to which the retrieved user belongs.
local	boolean	Value of <i>true</i> if the user is a local user; <i>false</i> if an LDAP user.
locale	string	Locale of the retrieved user.
locked	boolean	Value of <i>true</i> if the user is currently locked out; <i>false</i> if not.
roleAssignments	roleAssignmentDataObj	Role to associate with the user at the global level. See <code>getAllRoles()</code> , <code>getRole()</code> , and <code>getAllPermissions()</code> . Zero or more role associations allowed. If updating role assignments, respecify any global type roles that you want to retain.
superUser	boolean	Value of <i>true</i> for the built-in Coverity Connect admin user. Otherwise, <i>false</i> .
userModified	string	Name of the user who last updated the user record.
username	string	User name that matches the request.

### 3.6.69. Complex Type: userFilterSpecDataObj

#### 3.6.69.1. Description

Filter properties used to return a matching set of user records.

#### 3.6.69.2. Derived By

Restricting anyType

#### 3.6.69.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
assignable	boolean	Set to <i>true</i> to retrieve only those users who can own software issues; <i>false</i> to retrieve only those who cannot. Otherwise, do not set.

Component	Type	Description
disabled	boolean	Set to <i>true</i> to retrieve <i>disabled</i> users only. Set to <i>false</i> to retrieve enabled users only. Otherwise, do not set.
groupsList	string	Name of user group to which the retrieved users must belong. Zero or more groups allowed.
includeDetails	boolean	Set to <i>false</i> to prevent the inclusion of role assignments and other user details in the response. Defaults to <i>true</i> .
ldap	boolean	Set to <i>true</i> to retrieve only LDAP users; <i>false</i> to retrieve only local users. Otherwise, do not set.
locked	boolean	Set to <i>true</i> to retrieve only those users who have been locked out; <i>false</i> to retrieve only unlocked users. Otherwise, do not set.
namePattern	string	Glob pattern that matches the user name of the users to retrieve.
projectIdDataObj	projectIdDataObj	Name of project to which the retrieved set of users must have a role association.
startId	long	Internal. Do not specify.

### 3.6.70. Complex Type: usersPageDataObj

#### 3.6.70.1. Description

Returned page of user records and count of records in the page.

#### 3.6.70.2. Derived By

Restricting anyType

#### 3.6.70.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
users	userDataObj	A user record. Zero or more can be returned.
totalNumberOfRecords	int	Total number of user records returned.

### 3.6.71. Complex Type: userSpecDataObj

#### 3.6.71.1. Description

Specification for a user.

### 3.6.71.2. Derived By

Restricting anyType

### 3.6.71.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
disabled	boolean	To prevent the user from logging in to the UI, set to <i>true</i> . Defaults to <i>false</i> for <code>createUser()</code> .
domain	serverDomainIdDataObj	For an LDAP user only, the LDAP domain of user.
email	string	Email address of the user.
familyName	string	Last name of the user.
givenName	string	First name of the user.
groupNames	groupIdDataObj	Name of an existing group to which the user should belong. Zero or more group associations allowed. <i>Supported when updating a user, not when creating a user.</i>
local	boolean	If an LDAP user, set to <i>false</i> . Defaults to <i>true</i> with <code>createUser()</code> , specifying a local (non-LDAP) user.
locale	string	The locale of the user. Defaults to <i>en-US</i> with <code>createUser()</code> .
locked	boolean	To lock out the new user, set to <i>true</i> . Unless password recovery is enabled through the UI, the administrator must reset the password for the locked out user before the user can log in. Defaults to <i>false</i> with <code>createUser()</code> .
password	string	Password for the user. Required with <code>createUser()</code> .
roleAssignments	roleAssignmentDataObj	Role to associate with the new user at the global level. See <code>getAllRoles()</code> , <code>getRole()</code> , and <code>getAllPermissions()</code> . Zero or more role associations are allowed. If updating role assignments, respecify any that you want to retain.
username	string	Required. Username for/of the user. Any capitalized (upper case) letters in the name will be converted to lower case when using <code>createUser()</code> . Maximum of one name specification is allowed.

Component	Type	Description
-----------	------	-------------

### 3.6.72. Complex Type: versionDataObj

#### 3.6.72.1. Description

Returns Coverity Connect version data.

#### 3.6.72.2. Derived By

Restricting anyType

#### 3.6.72.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
externalVersion	string	External version. Used for licensing purposes.
internalVersion	string	Internal version. Used for support purposes.

## 3.7. Simple Types: Configuration Service

### 3.7.1. Description

Simple Configuration Service types.

### 3.7.2. Simple Types

Name	Description
deleteSnapshotJobStatus	Returns the status of a preceeding deleteSnaphot() request.

### 3.7.3. Simple Type: deleteSnapshotJobStatus

#### 3.7.3.1. Description

Returns the status of a preceeding deleteSnaphot() request.

#### 3.7.3.2. Derived By

Restricting string

#### 3.7.3.3. Enumeration

Value	Description
QUEUED	Queued for deletion.

<b>Value</b>	<b>Description</b>
RUNNING	Deletion in progress.
SUCCEEDED	Deleted successfully.
FAILED	Deletion failed.

---

#### **3.7.3.4. Remarks**

See `getDeleteSnapshotJobInfo()`.

---

## Chapter 4. Defect Service Web Service

### Table of Contents

4.1. Name .....	125
4.2. Description .....	125
4.3. Type .....	125
4.4. See Also .....	125
4.5. Operations: Defect Service .....	125
4.6. Complex Types .....	147
4.7. Types and Constants .....	179

### 4.1. Name

Defect Service

### 4.2. Description

API for operations that pertain to CIDs and instances (occurrences) of software issues.

### 4.3. Type

SOAP

### 4.4. See Also

Operations , Complex Types

### 4.5. Operations: Defect Service

#### 4.5.1. Operations

Name	Description
getComponentMetricsForProject	Retrieve metrics on components associated with streams in a specified project.
getFileContents	Retrieve the Base64-encoded value of the zlib-compressed contents of a file that contains an instance of a CID.
getMergedDefectDetectionHistory	Retrieves detection history for a software issue. The return data is similar to the Detection History information in the Coverity Connect UI.
getMergedDefectHistory	Retrieve a date and time stamped list of changes to attributes used to triage a specified CID.
getMergedDefectsForProjectScope	Retrieve CIDs (filtered or unfiltered) that are in a specified project.
getMergedDefectsForSnapshotScope	Retrieve CIDs (filtered or unfiltered) that are in the current or specified snapshots. Optionally, perform snapshot comparisons.

Name	Description
getMergedDefectsForStreams	Retrieve the current attributes and other properties of CIDs (filtered or unfiltered) in a specified stream.
getStreamDefects	Retrieve instances of software issues for one or more CIDs.
getTrendRecordsForProject	Retrieve daily records on CIDs and source code in a project.
getTriageHistory	Retrieve the triage history for a software issue.
updateDefectInstanceProperties	Do not use this operation.
updateStreamDefects	Update the one or more attribute values for <i>all</i> instances of a CID found in a given stream. Note that this update will apply to all instances of the CID in all streams that share the same triage store.
updateTriageForCIDsInTriageStore	Update one or more attribute values for a CID in a specified triage store.

## 4.5.2. Operation: getComponentMetricsForProject

### 4.5.2.1. Name

getComponentMetricsForProject

### 4.5.2.2. Description

Retrieve metrics on components associated with streams in a specified project.

### 4.5.2.3. Parameters

projectId

**Type:** projectIdDataObj

Identifier for a project.

Field name	Type	Description
name	string	Required. Name of the project.

componentIds

**Type:** componentIdDataObj

Identifier for a component.

Field name	Type	Description
name	string	Name of a component in the project in the form <b>[componentMap]. [component]</b> (for example, <i>myComponentMap.myComponent</i> ) Multiple names allowed.

#### 4.5.2.4. Output (Literal)

The output of this operation is the argument `GetComponentMetricsForProjectResponse` having the structure defined by the following table.

Name	Type
return	componentMetricsDataObj

#### 4.5.3. Operation: getFileContents

##### 4.5.3.1. Name

`getFileContents`

##### 4.5.3.2. Description

Retrieve the Base64-encoded value of the zlib-compressed contents of a file that contains an instance of a CID.

##### 4.5.3.3. Parameters

`streamId`

**Type:** `streamIdDataObj`

Field name	Type	Description
<code>name</code>	<code>string</code>	Required. Name of the stream. You can specify one or more instances of <code>streamIdDataObj</code> . See the example.

`fileId`

**Type:** `fileIdDataObj`

Field name	Type	Description
<code>contentsMD5</code>	<code>string</code>	Required. MD5 checksum (a fingerprint or message digest) of the file contents. You can get the <code>contentsMD5</code> and <code>filePathname</code> for an instance of a CID by using <code>getStreamDefects()</code> with the <code>includeDefectInstances</code> filter set to true.
<code>filePathname</code>	<code>string</code>	Required. Path to the file that contains the instance of the CID. You can get the <code>contentsMD5</code> and <code>filePathname</code> for an instance of a CID by using <code>getStreamDefects()</code> with the <code>includeDefectInstances</code> filter set to true.

#### 4.5.3.4. Output (Literal)

The output of this operation is the argument `getFileContentsResponse` having the structure defined by the following table.

Name	Type
return	fileContentsDataObj

#### 4.5.4. Operation: `getMergedDefectDetectionHistory`

##### 4.5.4.1. Name

`getMergedDefectDetectionHistory`

##### 4.5.4.2. Description

Retrieves detection history for a software issue. The return data is similar to the Detection History information in the Coverity Connect UI.

##### 4.5.4.3. Parameters

`mergedDefectIdDataObj`

**Type:** `mergedDefectIdDataObj`

The `cid` and/or `merge key` for a software issue. At least one of them must be provided.

Field name	Type	Description
<code>cid</code>	long	CID.
<code>mergeKey</code>	string	Numeric key for a CID.

`streamIds`

**Type:** `streamIdDataObj`

Name of a stream in which the software issue occurs.

Field name	Type	Description
<code>name</code>	string	Required. Name of the stream. You can specify one or more instances of <code>streamIdDataObj</code> . See the example.

#### 4.5.4.4. Output (Literal)

The output of this operation is the argument `getMergedDefectDetectionHistoryResponse` having the structure defined by the following table.

Name	Type
return	defectDetectionHistoryDataObj

#### 4.5.5. Operation: getMergedDefectHistory

##### 4.5.5.1. Name

getMergedDefectHistory

##### 4.5.5.2. Description

Retrieve a date and time stamped list of changes to attributes used to triage a specified CID.

##### 4.5.5.3. Parameters

mergedDefectIdDataObj

**Type:** mergedDefectIdDataObj

Identifier for a software issue. You must pass a cid and/or mergeKey value.

Field name	Type	Description
cid	long	CID.
mergeKey	string	Numeric key for a CID.

streamIds

**Type:** streamIdDataObj

Identifier for a stream.

Field name	Type	Description
name	string	Required. Name of the stream. You can specify one or more instances of streamIdDataObj. See the example.

##### 4.5.5.4. Output (Literal)

The output of this operation is the argument getMergedDefectHistoryResponse having the structure defined by the following table.

Name	Type
return	defectChangeDataObj

##### 4.5.5.5. Remarks

#### 4.5.6. Operation: getMergedDefectsForProjectScope

##### 4.5.6.1. Name

getMergedDefectsForProjectScope

##### 4.5.6.2. Description

Retrieve CIDs (filtered or unfiltered) that are in a specified project.

#### 4.5.6.3. Parameters

projectId

**Type:** projectIdDataObj

Passes an identifier for the project.

Field name	Type	Description
name	string	Required. Name of the project.

filterSpec

**Type:** projectScopeDefectFilterSpecDataObj

Passes an optional filters on the results to return.

Field name	Type	Description
actionNameList	string	Name/value pairs for a list of attributes.
checkerCategoryList	string	List of checker categories.
checkerList	string	List of checkers.
checkerTypeList	string	List of checker types.
cidList	long	List of CIDs.
classificationNameList	string	Classification of the CID. Multiple classifications allowed.
componentIdExclude	boolean	If one or more component name filters is specified, set to <i>true</i> to exclude matching results from the specified components. Defaults to <i>false</i> , including the matches from the components in the results.
componentIdList	componentIdDataObj	Name of a component that contains the CID. Multiple components allowed.
cweList	long	Common Weakness Enumeration identifier of the type of issue found by the checker. Zero or more identifiers allowed.
firstDetectedBy	string	Value that helps identify the process by which an issue was <i>initially reported</i> to Coverity Connect: <b>COMMIT</b> (for issues initially reported through a commit process that yields a snapshot; appears as " <b>Snapshot</b> " in the UI), <b>PREVIEW</b> (for issues initially reported through a preview process, which does not produce a snapshot, for example, when Coverity Desktop invokes cov-run-desktop), or <b>API</b> (for issues initially

Field name	Type	Description
		<p>reported through a special, rarely used process). In each case, a CID for the issue is created.</p> <p>Notes:</p> <p>Preview issues that a developer fixes before pushing code changes to the source code repository will never have (or need) a snapshot. Preview issues left unfixed before they are pushed to the repository will typically undergo the server-based analysis and commit process. Therefore, these issues will receive a snapshot in Coverity Connect <i>after</i> they are initially reported. Whether fixed or left unfixed prior to the push to the source code repository, issues will be identified as Preview issues if they were initially reported through a preview process.</p>
firstDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the First Detected date of a CID. Example1:2013-03-18T12:42:19.384-07:00Example
firstDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the First Detected date of a CID. For an example, see firstDetectedEndDate.
fixTargetNameList	string	Fix target for the CID; a triage value for the CID. Multiple fix targets allowed.
impactNameList	string	Probable impact (High, Medium, or Low) of the issue found by the checker. Zero or more impact levels allowed.
issueKindList	string	Issue kind. Multiple issue kinds allowed.
lastDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.
lastDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.

Field name	Type	Description
legacyNameList	string	Legacy designation for the CID (true or false), a triage value for the CID. Built-in attribute. Defaults to false.
ownerNameList	string	Owner of the CID.
ownerNamePattern	string	Glob pattern matching the first or last name of the owner of a CID.
severityNameList	string	Severity of the CID; a triage value for the CID. Multiple severities allowed.
standardAttributeValueFilterMap	StandardAttributeValueFilterMap	StandardAttributeValueFilterMap
specificCid	String	Specific CID.

pageSpec

**Type:** pageSpecDataObj

Passes page specifications for results. The pageSize field is required.

Field name	Type	Description
pageSize	int	Required. Up to 5000 records per page.
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Name of the field to use for sorting results. Not all fields are supported. However, you can typically sort by a field that returns numeric results, such as cid and the date fields.
startIndex	int	Zero-based index of records to return. Defaults to 0.

#### 4.5.6.4. Output (Literal)

The output of this operation is the argument `getMergedDefectsForProjectScopeResponse` having the structure defined by the following table.

Name	Type
return	mergedDefectsPageDataObj

#### 4.5.7. Operation: `getMergedDefectsForSnapshotScope`

##### 4.5.7.1. Name

`getMergedDefectsForSnapshotScope`

#### 4.5.7.2. Description

Retrieve CIDs (filtered or unfiltered) that are in the current or specified snapshots. Optionally, perform snapshot comparisons.

#### 4.5.7.3. Parameters

projectId

**Type:** projectIdDataObj

Passes an identifier for a project.

Field name	Type	Description
name	string	Required. Name of the project.

filterSpec

**Type:** snapshotScopeDefectFilterSpecDataObj

Passes optional filters on the results to return.

Field name	Type	Description
actionNameList	string	Name/value pairs for a list of attributes.
attributeDefinitionValueFilterMatchSpecDataObj	attributeDefinitionValueFilterMatchSpecDataObj	Specification of an attribute value.
checkerCategoryList	string	List of checker categories.
checkerList	string	List of checkers.
checkerTypeList	string	List of checker types.
cidList	long	List of CIDs.
classificationNameList	string	Classification of the CID. Multiple classifications allowed.
componentIdExclude	boolean	If one or more component name filters is specified, set to true to exclude matching results from the specified components. Defaults to false, including the matches from the components in the results.
componentIdList	componentIdDataObj	Name of a component that contains the CID. Multiple components allowed.
cweList	long	Common Weakness Enumeration identifier of the type of issue found by the checker. Zero or more identifiers allowed.
externalReference	string	An external reference for a CID that is used by your company to identify the software issue. Corresponds to a field in the Coverity Connect triage pane.

Field name	Type	Description
fileName	string	A file name. Example: /test.c
firstDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the First Detected date of a CID. Example1: 2013-03-18T12:42:19.384-07:00 Example2: 2013-03-18
firstDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the First Detected date of a CID. For an example, see firstDetectedEndDate.
fixTargetNameList	string	Fix target for the CID; a triage value for the CID. Multiple fix targets allowed.
functionMergeName	string	Internal function name used as one of the criteria for merging separate occurrences of the same software issue, with the result that they are identified by the same CID.
functionName	string	Name of the function or method.
impactNameList	string	Probable impact (High, Medium, or Low) of the issue found by the checker. Zero or more impact levels allowed.
issueComparison	string	If set to <i>PRESENT</i> , returns overlapping CIDs in a snapshot comparison, that is, CIDs found in snapshot(s) to which both the <i>showSelector</i> and <i>compareSelector</i> values of the <i>snapshotScope</i> parameter (snapshotScopeSpecDataObj) apply. If set to <i>ABSENT</i> , returns CIDs that are present in the snapshot(s) to which the <i>showSelector</i> value applies but absent from those to which the <i>compareSelector</i> value applies. If not set, values are <i>PRESENT</i> and <i>ABSENT</i> .
issueKindList	string	Issue kind. Multiple issue kinds allowed.
lastDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.
lastDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the

Field name	Type	Description
		Last Detected date of a CID. For an example, see <code>firstDetectedEndDate</code> .
<code>legacyNameList</code>	string	Legacy designation for the CID (true or false), a triage value for the CID. Built-in attribute. Defaults to false.
<code>maxOccurrenceCount</code>	int	Maximum number of instances of software issues associated with a given CID. See <code>minOccurrenceCount</code> .
<code>mergeExtra</code>	string	Internal property used as one of the criteria for merging occurrences of an issue.
<code>mergeKey</code>	string	Internal signature used to merge separate occurrences of the same software issue and identify them all by the same CID.
<code>minOccurrenceCount</code>	int	Minimum number of instances of software issues associated with a given CID. See <code>maxOccurrenceCount</code> .
<code>ownerNameList</code>	string	Owner of the CID.
<code>ownerNamePattern</code>	string	Glob pattern matching the first or last name of the owner of a CID.
<code>severityNameList</code>	string	Severity of the CID; a triage value for the CID. Multiple severities allowed.
<code>standardAttributeValueFilterMap</code>	standardAttributeValueFilterMap	Standard attribute value filter map for a CID.
<code>statusNameList</code>	string	Status of the CID. Multiple statuses allowed.
<code>streamExcludeNameList</code>	streamIdDataObj	Identifier for a stream to exclude. Multiple streams are allowed. See <code>streamExcludeQualifier</code> .
<code>streamExcludeQualifier</code>	string	If set to <i>ANY</i> , the filter will exclude from the results CIDs found in each of the streams listed in the <code>streamExcludeNameList</code> field. If set to <i>ALL</i> , the filter will only exclude a CID if it is found in all listed streams. Valid values are <i>ANY</i> or <i>ALL</i> . Defaults to <i>ANY</i> .
<code>streamIncludeNameList</code>	streamIdDataObj	Identifier for a stream to include. Multiple streams are allowed. See <code>streamIncludeQualifier</code> .

Field name	Type	Description
streamIncludeQualifier	string	If set to <i>ANY</i> , the filter will return CIDs found in each of the streams listed in the <i>streamIncludeNameList</i> field. If set to <i>ALL</i> , the filter will only return a CID if it is found in all listed streams. Valid values are <i>ANY</i> or <i>ALL</i> . Defaults to <i>ANY</i> .

## pageSpec

**Type:** pageSpecDataObj

Specification for the page of results to return.

Field name	Type	Description
pageSize	int	Required. Up to 5000 records per page.
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Name of the field to use for sorting results. Not all fields are supported. However, you can typically sort by a field that returns numeric results, such as <i>cid</i> and the date fields.
startIndex	int	Zero-based index of records to return. Defaults to <i>0</i> .

## snapshotScope

**Type:** snapshotScopeSpecDataObj

Specification used for snapshot comparison. The Coverity Connect UI also provides snapshot comparison functionality.

Field name	Type	Description
compareOutdatedStreams	boolean	If set to <i>true</i> , includes outdated streams found in snapshots specified by <i>compareSelector</i> . If <i>false</i> , the default, only non-outdated streams are included.  See the note in <i>showOutdatedStreams</i> .
compareSelector	string	Snapshot ID or snapshot grammar value that is used to set the scope of snapshots to compare with the <i>showSelector</i> snapshot scope. For

Field name	Type	Description
showOutdatedStreams	boolean	<p>more information, see "Snapshot comparison" in the Coverity Connect Administration Guide.</p> <p>If set to <i>true</i>, includes outdated streams found in snapshots specified by <i>showSelector</i>. If <i>false</i>, the default, only non-outdated streams are included.</p> <p>Note that a user with proper RBAC permissions at the stream level can designate a stream as <i>outdated</i> to exclude the stream from Coverity Connect processes.</p>
showSelector	string	<p>Require Snapshot ID or snapshot grammar value that is used to set the scope of snapshots</p> <p>Default: <i>last()</i> which includes the latest snapshot of each stream in the project.</p> <p>See the Coverity Platform Administration Guide for details on the snapshot grammar.</p>

#### 4.5.7.4. Output (Literal)

The output of this operation is the argument `getMergedDefectsForSnapshotScopeResponse` having the structure defined by the following table.

Name	Type
return	<code>mergedDefectsPageDataObj</code>

### 4.5.8. Operation: `getMergedDefectsForStreams`

#### 4.5.8.1. Name

`getMergedDefectsForStreams`

#### 4.5.8.2. Description

Retrieve the current attributes and other properties of CIDs (filtered or unfiltered) in a specified stream.

#### 4.5.8.3. Parameters

`streamIds`

**Type:** `streamIdDataObj`

Identifier for a stream.

Field name	Type	Description
name	string	Required. Name of the stream. You can specify one or more instances of streamIdDataObj. See the example.

filterSpec

**Type:** mergedDefectFilterSpecDataObj

Optional filter properties used to match CIDs to return from the specified stream.

Field name	Type	Description
cidList	long	A CID. Multiple CIDs allowed.
filenamePatternList	string	Filename pattern for source code files that containing software issues associated with the CIDs. Up to 20 patterns allowed.
componentIdList	componentIdDataObj	Name of a component that contains the CID. Multiple components allowed.
statusNameList	string	Status of the CID. Multiple statuses allowed.
classificationNameList	string	Classification of the CID; a triage value for the CID. Multiple classifications allowed.
actionNameList	string	Name/value pairs for a list of attributes.
fixTargetNameList	string	Fix target for the CID; a triage value for the CID. Multiple fix targets allowed.
severityNameList	string	Severity of the CID; a triage value for the CID. Multiple severities allowed.
legacyNameList	string	Legacy designation for the CID ( <i>true</i> or <i>false</i> ); a triage value for the CID. Built-in attribute. Defaults to <i>false</i> .
ownerNameList	string	Owner of the CID.
checkerList	string	List of checkers.
cweList	int	Common Weakness Enumeration identifier of the type of issue.
checkerCategoryList	string	List of checker categories.
checkerTypeList	string	List of checker types.
impactList	string	List of values for issue impact.
issueKindList	string	Kind of issue identified by the CID.
attributeDefinitionValueFilterMap	attributeDefinitionValueFilterMapDataObj	Specification of an attribute value.
componentIdExclude	boolean	If one or more component name filters is specified, set to <i>true</i> to

Field name	Type	Description
		exclude matching results from the specified components. Defaults to <i>false</i> , including the matches from the components in the results.
defectPropertyKey	string	Do not use this field. The API does not process these values.
defectPropertyPattern	string	Do not use this field. The API does not process these values.
externalReferencePattern	string	Glob pattern matching the value of an Ext. Reference attribute value.
firstDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the First Detected date of a CID.  <b>Example1:</b> 2013-03-18T12:42:19.384-07:00  <b>Example2:</b> 2013-03-18
firstDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the First Detected date of a CID.  For an example, see firstDetectedEndDate.
functionNamePattern	string	Glob pattern matching the name of the function (or method) associated with a CID.
lastDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Detected date of a CID.  For an example, see firstDetectedEndDate.
lastDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the Last Detected date of a CID.  For an example, see firstDetectedEndDate.
lastFixedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Fixed date of a CID.

Field name	Type	Description
lastFixedStartDate	dateTime	<p>For an example, see firstDetectedEndDate.</p> <p>Starting date (and optionally, time) for the date range matching the Last Fixed date of a CID.</p>
lastTriagedEndDate	dateTime	<p>For an example, see firstDetectedEndDate.</p> <p>Ending date (and optionally, time) for the date range matching the Last Triaged date of a CID.</p>
lastTriagedStartDate	dateTime	<p>For an example, see firstDetectedEndDate.</p> <p>Starting date (and optionally, time) for the date range matching the Last Triaged date of a CID.</p>
maxCid	long	<p>For an example, see firstDetectedEndDate.</p> <p>Upper numeric bound of CIDs to retrieve. For example, no greater than CID 25000.</p>
maxOccurrenceCount	int	<p>See minCid.</p> <p>Maximum number of instances of software issues associated with a given CID.</p>
mergedDefectIdDataObjs	mergedDefectIdDataObj	<p>See minOccurrenceCount.</p> <p>Identifier for a software issue.</p>
minCid	long	<p>Multiple specifications are allowed.</p> <p>Lower numeric bound of CIDs to retrieve. For example, no smaller than CID 24500.</p>
minOccurrenceCount	int	<p>See maxCid.</p> <p>Minimum number of instances of software issues associated with a given CID.</p> <p>See maxOccurrenceCount.</p>

Field name	Type	Description
ownerNamePattern	string	Glob pattern matching the first or last name of the owner of a CID.
snapshotComparisonField	string	Specifies the snapshot used for comparison filtering.
standardAttributeFilterMap	standardAttributeFilterMap	Specification of a standard attribute.
streamExcludeNameList	streamIdDataObj	Identifier for a stream to exclude. Multiple streams are allowed.
streamExcludeQualifier	string	Specifies whether the filter works for 'ANY' or 'ALL' of the excluded streams.
streamIncludeNameList	streamIdDataObj	Identifier for a stream to include. Multiple streams are allowed.
streamIncludeQualifier	string	Specifies whether the filter works for 'ANY' or 'ALL' of the included streams.

## pageSpec

**Type:** pageSpecDataObj

Specification for the page of results to return. The pageSize field is required.

Field name	Type	Description
pageSize	int	Required. Up to 5000 records per page.
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Name of the field to use for sorting results. Not all fields are supported. However, you can typically sort by a field that returns numeric results, such as cid and the date fields.
startIndex	int	Zero-based index of records to return. Defaults to 0.

## snapshotScope

**Type:** snapshotScopeSpecDataObj

Optional parameter for adjusting the snapshot scope.

Field name	Type	Description
compareOutdatedStreams	boolean	If set to <i>true</i> , includes outdated streams found in snapshots specified by <i>compareSelector</i> . If <i>false</i> , the default, only non-outdated streams are included.

Field name	Type	Description
compareSelector	string	See the note in <i>showOutdatedStreams</i> . Snapshot ID or snapshot grammar value that is used to set the scope of snapshots to compare with the <i>showSelector</i> snapshot scope. For more information, see "Snapshot comparison" in the Coverity Connect Administration Guide.
showOutdatedStreams	boolean	If set to <i>true</i> , includes outdated streams found in snapshots specified by <i>showSelector</i> . If <i>false</i> , the default, only non-outdated streams are included.  Note that a user with proper RBAC permissions at the stream level can designate a stream as <i>outdated</i> to exclude the stream from Coverity Connect processes.
showSelector	string	Require Snapshot ID or snapshot grammar value that is used to set the scope of snapshots  Default: <i>last()</i> which includes the latest snapshot of each stream in the project.  See the Coverity Platform Administration Guide for details on the snapshot grammar.

#### 4.5.8.4. Output (Literal)

The output of this operation is the argument `getMergedDefectsForStreamsResponse` having the structure defined by the following table.

Name	Type
return	<code>mergedDefectsPageDataObj</code>

#### 4.5.9. Operation: `getStreamDefects`

##### 4.5.9.1. Name

`getStreamDefects`

##### 4.5.9.2. Description

Retrieve instances of software issues for one or more CIDs.

#### 4.5.9.3. Parameters

mergedDefectIdDataObjs (one or more)

**Type:** mergedDefectIdDataObj

Field name	Type	Description
cid	long	CID.
mergeKey	string	Numeric key for a CID.

filterSpec

**Type:** streamDefectFilterSpecDataObj

Field name	Type	Description
defectStateEndDate	dateTime	Ending date (and optionally, time) for the CIDs to return.
defectStateStartDate	dateTime	Starting date (and optionally, time) for the CIDs to return.
includeDefectInstances	boolean	Set to <i>true</i> for data on each instance of software issue, including the ID. Defaults to <i>false</i> .
includeHistory	boolean	Set to <i>true</i> for historical triage data on each instance of the software issue.
streamIdList	streamIdDataObj	Identifier for a stream. Multiple streams allowed.

#### 4.5.9.4. Output (Literal)

The output of this operation is the argument `getStreamDefectsResponse` having the structure defined by the following table.

Name	Type
return	streamDefectDataObj

#### 4.5.10. Operation: `getTrendRecordsForProject`

##### 4.5.10.1. Name

`getTrendRecordsForProject`

##### 4.5.10.2. Description

Retrieve daily records on CIDs and source code in a project.

### 4.5.10.3. Parameters

projectId

**Type:** projectIdDataObj

Field name	Type	Description
name	string	Required. Name of the project.

filterSpec

**Type:** projectTrendRecordFilterSpecDataObj

Field name	Type	Description
endDate	dateTime	End date (and optionally, time) for the set of records to retrieve.
startDate	dateTime	Start date (and optionally, time) for the set of records to retrieve.

### 4.5.10.4. Output (Literal)

The output of this operation is the argument `getTrendRecordsForProjectResponse` having the structure defined by the following table.

Name	Type
return	projectMetricsDataObj

## 4.5.11. Operation: getTriageHistory

### 4.5.11.1. Name

getTriageHistory

### 4.5.11.2. Description

Retrieve the triage history for a software issue.

### 4.5.11.3. Parameters

mergedDefectIdDataObj

**Type:** mergedDefectIdDataObj

Specifies a CID and/or merge key for a software issue.

Field name	Type	Description
cid	long	CID.

Field name	Type	Description
mergeKey	string	Numeric key for a CID.

triageStoreIds

**Type:** triageStoreIdDataObj

Identifier for a triage store.

Field name	Type	Description
name	string	Required. Name of the triage store.

#### 4.5.11.4. Output (Literal)

The output of this operation is the argument getTriageHistoryResponse having the structure defined by the following table.

Name	Type
return	trriageHistoryDataObj

#### 4.5.12. Operation: updateDefectInstanceProperties

##### 4.5.12.1. Name

updateDefectInstanceProperties

##### 4.5.12.2. Description

Do not use this operation.

##### 4.5.12.3. Parameters

defectInstanceId

**Type:** defectInstanceIdDataObj

Field name	Type	Description
id	long	Do not use this field. The API does not process these values.

properties

**Type:** propertySpecDataObj

Field name	Type	Description
key	string	Do not use this field. The API does not process these values.

Field name	Type	Description
value	string	Do not use this field. The API does not process.

#### 4.5.12.4. Remarks

Users should not use this operation or its fields.

### 4.5.13. Operation: updateStreamDefects

#### 4.5.13.1. Name

updateStreamDefects

#### 4.5.13.2. Description

Update the one or more attribute values for *all* instances of a CID found in a given stream. Note that this update will apply to all instances of the CID in all streams that share the same triage store.

#### 4.5.13.3. Parameters

streamDefectIds

**Type:** streamDefectIdDataObj

Field name	Type	Description
defectTriageId	long	Internal value for the last known triage ID. This ID changes when developers triage the issue that is associated with the <i>id</i> .
defectTriageVerNum	int	Internal value for the last known triage version. This number changes when developers triage the issue that is associated with the <i>id</i> .
id	long	Internal identifier for the software issue within the context of the stream.
verNum	int	Version number associated with the <i>id</i> .

defectStateSpec

**Type:** defectStateSpecDataObj

Field name	Type	Description
defectStateAttributeValues	defectStateAttributeValueDataObj	Attribute name/value pair. One or more pairs required.

#### 4.5.13.4. Remarks

Limit of 500 streamDefectIds. To retrieve a list of *streamDefectIdDataObj* values for a CID, see `getStreamDefects()`.

#### 4.5.14. Operation: updateTriageForCIDsInTriageStore

##### 4.5.14.1. Name

updateTriageForCIDsInTriageStore

##### 4.5.14.2. Description

Update one or more attribute values for a CID in a specified triage store.

##### 4.5.14.3. Parameters

trriageStore

**Type:** triageStoreIdDataObj

Identifier for a triage store.

Field name	Type	Description
name	string	Required. Name of the triage store.

mergedDefectIdDataObjs

**Type:** mergedDefectIdDataObj

Identifier for a software issue. A cid and/or mergeKey is required.

Field name	Type	Description
cid	long	CID.
mergeKey	string	Numeric key for a CID.

defectState

**Type:** defectStateSpecDataObj

An triage attribute name/value pair.

Field name	Type	Description
defectStateAttributeValues	defectStateAttributeValueDataObj	Attribute name/value pair. One or more pairs required.

## 4.6. Complex Types: Defect Service

### 4.6.1. Description

Defect Service data objects.

## 4.6.2. Complex Types

Name	Description
attributeDefinitionIdDataObj	Identifier for an attribute.
attributeDefinitionValueFilterMapDataObj	Filter for one or more attribute values.
attributeValueIdDataObj	Identifier for the attribute.
componentIdDataObj	Identifier for the component.
componentMetricsDataObj	Metrics for a component.
CovRemoteServiceException	Error code and message.
defectChangeDataObj	Returns data on the CID.
defectDetectionHistoryDataObj	Returns detection history for a software issue in a snapshot.
defectInstanceDataObj	Returns data on an instance of a software issue.
defectInstanceIdDataObj	Identifier for an instance of a software issue.
defectStateAttributeValueDataObj	Name/value pair for an attribute value.
defectStateDataObj	Returns historical triage data on a software issue.
defectStateSpecDataObj	Updated attribute values.
eventDataObj	Returns data about one or more events that contributed to a software issue. An event is a message associated with a particular line of code that explains some part of a software issue.
fieldChangeDataObj	Returns the old and new value of an attribute.
fileContentsDataObj	Returns the contents of a source code file.
fileIdDataObj	Contents of and path to a source code file.
functionInfoDataObj	Returns data on a function or method.
localizedValueDataObj	Object containaing a display name and ID for various issue attributes.
mergedDefectDataObj	Returns data on a CID.
mergedDefectFilterSpecDataObj	Filter on the CIDs to return.
mergedDefectIdDataObj	Specification that identifies a software issue.
mergedDefectsPageDataObj	Returns data on the requested CIDs.
pageSpecDataObj	Specification for the page of records to return.
projectIdDataObj	Identifier for a project.
projectMetricsDataObj	Triage and source code data on CIDs in a project.
projectScopeDefectFilterSpecDataObj	Passes optional filter properties matching the issues to return.
projectTrendRecordFilterSpecDataObj	Filter for project trend records to return.
propertyDataObj	A key/value pair for a property of an instance of a software issue.
propertySpecDataObj	A key/value pair for a property of an instance of a software issue.
snapshotScopeDefectFilterSpecDataObj	Filter on the snapshots to return.

Name	Description
snapshotScopeSpecDataObj	Specification used for snapshot comparison.
standardAttributeIdDataObj	Identifier for a standard attribute.
standardAttributeValueFilterMapDataObj	Filter for one or more standard attribute values.
standardAttributeValueIdDataObj	Identifier for a standard attribute value.
streamDefectDataObj	Returns data on a CID within the context of a stream.
streamDefectFilterSpecDataObj	Filter used to return matching software issues within the scope of one or more streams.
streamDefectIdDataObj	Identifier for a software issue within the scope of a stream.
streamIdDataObj	Identifier for the stream.
triageHistoryDataObj	Data object that capture state of triage attributes for a CID at a given point in time. For example, if a developer designates a new CID as a bug, the triage state changes.
triageStoreIdDataObj	Identifier for the triage store.

#### 4.6.3. Complex Type: attributeDefinitionIdDataObj

##### 4.6.3.1. Description

Identifier for an attribute.

##### 4.6.3.2. Derived By

Restricting anyType

##### 4.6.3.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
name	string	Name of the attribute.

#### 4.6.4. Complex Type: attributeDefinitionValueFilterMapDataObj

##### 4.6.4.1. Description

Filter for one or more attribute values.

##### 4.6.4.2. Derived By

Restricting anyType

##### 4.6.4.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
attributeDefinitionId	attributeDefinitionIdDataObj	Identifier for the attribute to filter.
attributeValueIds	attributeValueIdDataObj	Value of the attribute to filter. Multiple values allowed.

#### 4.6.5. Complex Type: attributeValueIdDataObj

##### 4.6.5.1. Description

Identifier for the attribute.

##### 4.6.5.2. Derived By

Restricting anyType

##### 4.6.5.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Name for the attribute.

#### 4.6.6. Complex Type: componentIdDataObj

##### 4.6.6.1. Description

Identifier for the component.

##### 4.6.6.2. Derived By

Restricting anyType

##### 4.6.6.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Name of a component in the project in the form <b>[componentMap].[component]</b> (for example, <i>myComponentMap.myComponent</i> ) Multiple names allowed.

### 4.6.7. Complex Type: componentMetricsDataObj

#### 4.6.7.1. Description

Metrics for a component.

#### 4.6.7.2. Derived By

Restricting anyType

#### 4.6.7.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
blankLineCount	int	Number of blank lines in the source code files that make up the component.
codeLineCount	int	Number of lines of code in the source code files that make up the component.
commentLineCount	int	Number of lines of comments in the source code files that make up the component.
componentId	componentIdDataObj	Name of the component.
dismissedCount	int	Number of CIDs in the component that developers have dismissed (classified as False Positive or Intentional).
fixedCount	int	Number of CIDs in the component that developers have fixed.
metricsDate	dateTime	Date and time stamp for the metric.
newCount	int	Number of CIDs that remain unclassified.
outstandingCount	int	Number of CIDs that remain. Does not include CIDs that have been resolved (fixed or dismissed).
totalCount	int	Total number of CIDs found in the component (whether outstanding or resolved).
triagedCount	int	Number of outstanding CIDs that have been triaged.

### 4.6.8. Complex Type: CovRemoteServiceException

#### 4.6.8.1. Description

Error code and message.

#### 4.6.8.2. Derived By

Restricting anyType

#### 4.6.8.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
errorCode	int	Error code.
message	string	Message associated with the error code.

---

#### 4.6.9. Complex Type: defectChangeDataObj

##### 4.6.9.1. Description

Returns data on the CID.

##### 4.6.9.2. Derived By

Restricting anyType

##### 4.6.9.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
affectedStreams	streamIdDataObj	Name of the stream that contains the specified CID.
attributeChanges	fieldChangeDataObj	Changes to an attribute for the CID.
comments	string	Comment on the CID.
dateModified	dateTime	Date and time the CID was created or modified.
userModified	string	Username of user who triaged the CID.

---

#### 4.6.10. Complex Type: defectDetectionHistoryDataObj

##### 4.6.10.1. Description

Returns detection history for a software issue in a snapshot.

##### 4.6.10.2. Derived By

Restricting anyType

#### 4.6.10.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
defectDetection	string	Detection status of the CI FIRST_DETECTED or LAST_DETECTED. Note that this field might be omitted if neither value applies, for example, in the case that a software issue appears and disappears from the analysis results several times.
detection	dateTime	Date and time that the issue was detected.
inCurrentSnapshot	boolean	True if the issue is in the current snapshot, false if not.
snapshotId	long	Identifier for the snapshot. Note that this field does not appear if inCurrentSnapshot is <i>true</i> .
streams	streamIdDataObj	Name of a stream.
userName	string	The username of the person who committed the analysis results to Coverity Connect.

#### 4.6.11. Complex Type: defectInstanceDataObj

##### 4.6.11.1. Description

Returns data on an instance of a software issue.

##### 4.6.11.2. Derived By

Restricting anyType

##### 4.6.11.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
events	eventDataObj	Data on an event that contributed to a software issue. Multiple events are possible.
properties	propertyDataObj	A key/value pair for a property of an instance of a software issue.
category	localizedValueDataObj	Categorization of the software issue.
checkerName	string	Name of the checker that found the issue.
component	string	

Component	Type	Description
cwe	int	Common Weakness Enumeration identifier of the type of the issue.
domain	string	Domain of the checker.
eventSetCaptions	string	Description available for occurrences of the software issue. Such captions appear in the UI above enumerated instances.
extra	string	Internal. Used to associate instance of the same CID.
function	functionInfoDataObj	Data on the function or method that contains the software issue.
id	defectInstanceCidDataObj	Identifier for an instance of a software issue.
impact	localizedValueDataObj	Probable impact of the software issue.
issueKinds	localizedValueDataObj	Kind of the issue.
localEffect	string	Local effect of the issue.
longDescription	string	Full description of the software issue.
subcategory	string	
type	localizedValueDataObj	Name of the issue type.

#### 4.6.12. Complex Type: defectInstanceCidDataObj

##### 4.6.12.1. Description

Identifier for an instance of a software issue.

##### 4.6.12.2. Derived By

Restricting anyType

##### 4.6.12.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
id	long	Do not use this field. The API does not process these values.

#### 4.6.13. Complex Type: defectStateAttributeValueDataObj

##### 4.6.13.1. Description

Name/value pair for an attribute value.

#### 4.6.13.2. Derived By

Restricting anyType

#### 4.6.13.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
attributeDefinitionId	attributeDefinitionIdDataObj	Identifier for an attribute.
attributeValueId	attributeValueIdDataObj	Value of the attribute.

#### 4.6.14. Complex Type: defectStateDataObj

##### 4.6.14.1. Description

Returns historical triage data on a software issue.

##### 4.6.14.2. Derived By

Restricting anyType

##### 4.6.14.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
dateCreated	dateTime	Date and time that one or more attribute values were created or updated.
defectStateAttributeValues	defectStateAttributeValueDataObj	Set of attribute/value pairs for a software issue.
userCreated	string	Username of the user who updated the attribute values.

#### 4.6.15. Complex Type: defectStateSpecDataObj

##### 4.6.15.1. Description

Updated attribute values.

##### 4.6.15.2. Derived By

Restricting anyType

#### 4.6.15.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
defectStateAttributeValues	defectStateAttributeValueDataObj	Attribute name/value pair. One or more pairs required.

#### 4.6.16. Complex Type: eventDataObj

##### 4.6.16.1. Description

Returns data about one or more events that contributed to a software issue. An event is a message associated with a particular line of code that explains some part of a software issue.

##### 4.6.16.2. Derived By

Restricting anyType

##### 4.6.16.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
events	eventDataObj	Container for event data. Multiple events for a single CID are possible.
eventDescription	string	Description of the event.
eventKind	string	The event kind.
eventNumber	int	Internal 1-based sequence number for ordering path events. Path events show the flow of control through a method or function along a path where the software issue occurs.
eventSet	int	Internal integer that is non-zero for multi-event displays of software issues.
eventTag	string	Short identifier for an occurrence (instance) of a software issue. Used in the UI.
fileId	fileIdDataObj	Identifier for the file that contains the software issue.
id	long	Identifier for the software issue.
lineNumber	int	Line number on which the software issue occurs in the source file.

Component	Type	Description
main	boolean	Value of <i>true</i> for the main event of a software issue.
moreInformationId	string	In the unlikely event that no event is designated as the main one, use the last event in the sequence as the main event. Internal. ID to documentation that provides background on a potential security vulnerability or other software issue. The link to this information appears in the UI.
pathCondition	string	Internal. Text of a condition tested in the code. Relates to the flow of control through a function along a path where a software issue occurs.
polarity	boolean	Internal.

#### 4.6.17. Complex Type: fieldChangeDataObj

##### 4.6.17.1. Description

Returns the old and new value of an attribute.

##### 4.6.17.2. Derived By

Restricting anyType

##### 4.6.17.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
fieldName	string	Name of the attribute.
newValue	string	New value for the attribute.
oldValue	string	Old value for the attribute.

#### 4.6.18. Complex Type: fileContentsDataObj

##### 4.6.18.1. Description

Returns the contents of a source code file.

##### 4.6.18.2. Derived By

Restricting anyType

**4.6.18.3. Content Model**

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
contents	base64Binary	Base64-encoded contents of the zlib-compressed file that contains the instance of the CID.
fileId	fileIdDataObj	ID of the file that contains the instance of the CID.

**4.6.19. Complex Type: fileIdDataObj****4.6.19.1. Description**

Contents of and path to a source code file.

**4.6.19.2. Derived By**

Restricting anyType

**4.6.19.3. Content Model**

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
contentsMD5	string	Required. MD5 checksum (a fingerprint or message digest) of the file contents. You can get the contentsMD5 and filePathname for an instance of a CID by using <code>getStreamDefects()</code> with the <code>includeDefectInstances</code> filter set to true.
filePathname	string	Required. Path to the file that contains the instance of the CID. You can get the contentsMD5 and filePathname for an instance of a CID by using <code>getStreamDefects()</code> with the <code>includeDefectInstances</code> filter set to true.

**4.6.20. Complex Type: functionInfoDataObj****4.6.20.1. Description**

Returns data on a function or method.

**4.6.20.2. Derived By**

Restricting anyType

#### 4.6.20.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
fileId	fileIdDataObj	Identifier for the file in which the function or method occurs.
functionDisplayName	string	Name of the function or method that is displayed in the UI.
functionMangledName	string	Mangled name of the function or method.
functionMergeName	string	

#### 4.6.21. Complex Type: localizedValueDataObj

##### 4.6.21.1. Description

Object containaing a display name and ID for various issue attributes.

##### 4.6.21.2. Derived By

Restricting anyType

##### 4.6.21.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
displayName	string	Localized value suitable for displaying to the user.
name	string	ID value suitable for passing as a filter.

#### 4.6.22. Complex Type: mergedDefectDataObj

##### 4.6.22.1. Description

Returns data on a CID.

##### 4.6.22.2. Derived By

Restricting anyType

#### 4.6.22.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
checkerName	string	Name of the checker that found the software issue associated with the CID.
cid	long	CID to which one or more instances of a software issue are associated.
componentName	string	Name of the component to which the issue belongs.
cwe	int	Common Weakness Enumeration identifier of the type of the issue.
defectStateAttributeValues	defectStateAttributeValueDataObj	List of attribute/value pairs associated with the software issue.
displayCategory	string	Name of the issue category.
displayImpact	string	Probable impact of the issue.
displayIssueKind	string	Issue kind.
displayType	string	Name of the issue type.
domain	string	Domain of the issue.
filePathname	string	Path to the file that contains the source file that contains the software issue.
firstDetected	dateTime	Date and time when the issue was first detected.
firstDetectedBy	string	Value that helps identify the process by which an issue was <i>initially reported</i> to Coverity Connect: <b>COMMIT</b> (for issues initially reported through a commit process that yields a snapshot; appears as " <b>Snapshot</b> " in the UI), <b>PREVIEW</b> (for issues initially reported through a preview process, which does not produce a snapshot, for example, when Coverity Desktop invokes cov-run-desktop), or <b>API</b> (for issues initially reported through a special, rarely used process). In each case, a CID for the issue is created.
firstDetectedDescription	string	First description of the snapshot (containing the CID) submitted to the stream. Submitted by using the optional --description option of cov-commit-defects command.
firstDetectedSnapshotId	long	Identifier of the snapshot in which the CID was first detected.

---

<b>Component</b>	<b>Type</b>	<b>Description</b>
firstDetectedStream	string	Stream in which the CID was first detected.
firstDetectedTarget	string	Target platform (for example, i386) of the source code in the snapshot in which the CID was first detected. Submitted by using the optional --target option of cov-commit-defects command.
firstDetectedVersion	string	Typically the version of the source code in the snapshot in which the CID was first detected. Submitted by using the optional --version option of cov-commit-defects command.
functionDisplayName	string	Name of the method or function associated with the CID as it appears in the UI.
functionMergeName	string	
functionName	string	Name of the method or function that is associated with the CID.
issueKind	string	Internal value for issue kind.
lastDetected	dateTime	Date and time when the CID was last detected.
lastDetectedDescription	string	Last description of the snapshot (containing the issue) submitted to the stream. Submitted by using the optional --description option of cov-commit-defects command.
lastDetectedSnapshotId	long	Identifier of the snapshot in which the CID was last detected.
lastDetectedStream	string	Stream in which the CID was last detected.
lastDetectedTarget	string	Target platform (for example, i386) of the source code in the snapshot in which the CID was last detected. Submitted by using the optional --target option of cov-commit-defects command.
lastDetectedVersion	string	Typically the version of the source code in the snapshot in which the CID was last detected. Submitted by using the optional --version option of cov-commit-defects command.
lastFixed	dateTime	Date and time when the CID was fixed (no longer occurred in a snapshot).
lastTriaged	dateTime	Date and time when the CID was last triaged.
mergeKey	string	Internal key used to associate instances of a software issue into a CID.
occurrenceCount	int	Number of instances of the software issue that are associated with the CID.

---

### 4.6.23. Complex Type: mergedDefectFilterSpecDataObj

#### 4.6.23.1. Description

Filter on the CIDs to return.

#### 4.6.23.2. Derived By

Restricting anyType

#### 4.6.23.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
cidList	long	A CID. Multiple CIDs allowed.
filenamePatternList	string	Filename pattern for source code files that containing software issues associated with the CIDs. Up to 20 patterns allowed.
componentIdList	componentIdDataObj	Name of a component that contains the CID. Multiple components allowed.
statusNameList	string	Status of the CID. Multiple statuses allowed.  By default, CID information is returned for the status types <code>New</code> , <code>Triaged</code> , and <code>Dismissed</code> .
classificationNameList	string	Classification of the CID; a triage value for the CID. Multiple classifications allowed.
actionNameList	string	Name/value pairs for a list of attributes.
fixTargetNameList	string	Fix target for the CID; a triage value for the CID. Multiple fix targets allowed.
severityNameList	string	Severity of the CID; a triage value for the CID. Multiple severities allowed.
legacyNameList	string	Legacy designation for the CID ( <i>true</i> or <i>false</i> ); a triage value for the CID. Built-in attribute. Defaults to <i>false</i> .
ownerNameList	string	Owner of the CID.
checkerList	string	List of checkers.
cweList	int	Common Weakness Enumeration identifier of the type of issue.
checkerCategoryList	string	List of checker categories.
checkerTypeList	string	List of checker types.
impactList	string	List of values for issue impact.

Component	Type	Description
issueKindList	string	Kind of issue identified by the CID.
attributeDefinitionValueFilterMap	Map	Map of an attribute value.
componentIdExclude	boolean	If one or more component name filters is specified, set to <i>true</i> to exclude matching results from the specified components. Defaults to <i>false</i> , including the matches from the components in the results.
defectPropertyKey	string	Do not use this field. The API does not process these values.
defectPropertyPattern	string	Do not use this field. The API does not process these values.
externalReferencePattern	string	Glob pattern matching the value of an Ext. Reference attribute value.
firstDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the First Detected date of a CID.  <b>Example1:</b> 2013-03-18T12:42:19.384-07:00 <b>Example2:</b> 2013-03-18
firstDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the First Detected date of a CID.  For an example, see firstDetectedEndDate.
functionNamePattern	string	Glob pattern matching the name of the function (or method) associated with a CID.
lastDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Detected date of a CID.  For an example, see firstDetectedEndDate.
lastDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the Last Detected date of a CID.  For an example, see firstDetectedEndDate.
lastFixedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Fixed date of a CID.  For an example, see firstDetectedEndDate.
lastFixedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the Last Fixed date of a CID.  For an example, see firstDetectedEndDate.
lastTriagedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Triaged date of a CID.

Component	Type	Description
lastTriagedStartDate	dateTime	For an example, see firstDetectedEndDate. Starting date (and optionally, time) for the date range matching the Last Triaged date of a CID.
maxCid	long	For an example, see firstDetectedEndDate. Upper numeric bound of CIDs to retrieve. For example, no greater than CID 25000.
maxOccurrenceCount	int	See minCid. Maximum number of instances of software issues associated with a given CID.
mergedDefectIdDataObjs	mergedDefectIdDataObj	See minOccurrenceCount. Identifier for a software issue.
minCid	long	Multiple specifications are allowed. Lower numeric bound of CIDs to retrieve. For example, no smaller than CID 24500.
minOccurrenceCount	int	See maxCid. Minimum number of instances of software issues associated with a given CID.
ownerNamePattern	string	See maxOccurrenceCount. Glob pattern matching the first or last name of the owner of a CID.
snapshotComparisonField	string	Specifies the snapshot used for comparison filtering.
streamExcludeNameList	streamIdDataObj	Identifier for a stream to exclude. Multiple streams are allowed.
streamExcludeQualifier	string	Specifies whether the filter works for 'ANY' or 'ALL' of the excluded streams.
streamIncludeNameList	streamIdDataObj	Identifier for a stream to include. Multiple streams are allowed.
streamIncludeQualifier	string	Specifies whether the filter works for 'ANY' or 'ALL' of the included streams.

#### 4.6.24. Complex Type: mergedDefectIdDataObj

##### 4.6.24.1. Description

Specification that identifies a software issue.

#### 4.6.24.2. Derived By

Restricting anyType

#### 4.6.24.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
cid	long	CID.
mergeKey	string	Numeric key for a CID.

---

#### 4.6.25. Complex Type: mergedDefectsPageDataObj

##### 4.6.25.1. Description

Returns data on the requested CIDs.

##### 4.6.25.2. Derived By

Restricting anyType

##### 4.6.25.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
mergedDefectIds	mergedDefectIdDataObj	Merge key and/or CID for a defect.
mergedDefects	mergedDefectDataObj	Properties of a CID. See the responses in the Example, below.
totalNumberOfRecords	int	Total number of records returned by the request.

---

#### 4.6.26. Complex Type: pageSpecDataObj

##### 4.6.26.1. Description

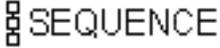
Specification for the page of records to return.

##### 4.6.26.2. Derived By

Restricting anyType

##### 4.6.26.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
		
pageSize	int	Required. Up to 1000 records per page.
sortAscending	boolean	Set to <i>false</i> to return records in reverse alphabetical or numerical order. Defaults to <i>true</i> .
sortField	string	Name of the field to use for sorting results. Not all fields are supported. However, you can typically sort by a field that returns numeric results, such as cid and the date fields.
startIndex	int	Zero-based index of records to return. Defaults to 0.

#### 4.6.27. Complex Type: projectIdDataObj

##### 4.6.27.1. Description

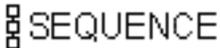
Identifier for a project.

##### 4.6.27.2. Derived By

Restricting anyType

##### 4.6.27.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
		
name	string	Required. Name of the project.

#### 4.6.28. Complex Type: projectMetricsDataObj

##### 4.6.28.1. Description

Triage and source code data on CIDs in a project.

##### 4.6.28.2. Derived By

Restricting anyType

##### 4.6.28.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
blankLineCount	int	Number of blank lines in the analyzed source code files.
codeLineCount	int	Number of lines of code in the analyzed source code files.
commentLineCount	int	Number of lines of comments in the analyzed source code files.
dismissedCount	int	Number of CIDs with the Status of Dismissed.
fixedCount	int	Number of CIDs with the Status of Fixed.
inspectedCount	int	Number of CIDs that have been triaged or fixed by developers.
metricsDate	dateTime	Date and time of the record.
newCount	int	Number of CIDs with a Classification of Unclassified. Also called <i>uninspected</i> issues.
outstandingCount	int	Number of CIDs with a Classification of Unclassified, Pending, or Bug.
projectId	projectIdDataObj	Identifier of the project.
resolvedCount	int	Number of CIDs with a Classification of Intentional or False Positive.
totalCount	int	Total number of CIDs.
triagedCount	int	Number of CIDs whose attributes have been triaged.

#### 4.6.29. Complex Type: projectScopeDefectFilterSpecDataObj

##### 4.6.29.1. Description

Passes optional filter properties matching the issues to return.

##### 4.6.29.2. Derived By

Restricting anyType

##### 4.6.29.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
actionNameList	string	Name/value pairs for a list of attributes.

Component	Type	Description
checkerCategoryList	string	List of checker categories.
checkerList	string	List of checkers.
checkerTypeList	string	List of checker types.
cidList	long	List of CIDs.
classificationNameList	string	Classification of the CID. Multiple classifications allowed.
componentIdExclude	boolean	If one or more component name filters is specified, set to <i>true</i> to exclude matching results from the specified components. Defaults to <i>false</i> , including the matches from the components in the results.
componentIdList	componentIdDataObj	Name of a component that contains the CID. Multiple components allowed.
cweList	long	Common Weakness Enumeration identifier of the type of issue found by the checker. Zero or more identifiers allowed.
firstDetectedBy	string	Value that helps identify the process by which an issue was <i>initially reported</i> to Coverity Connect: <b>COMMIT</b> (for issues initially reported through a commit process that yields a snapshot; appears as " <b>Snapshot</b> " in the UI), <b>PREVIEW</b> (for issues initially reported through a preview process, which does not produce a snapshot, for example, when Coverity Desktop invokes cov-run-desktop), or <b>API</b> (for issues initially reported through a special, rarely used process). In each case, a CID for the issue is created.

## Notes:

Preview issues that a developer fixes before pushing code changes to the source code repository will never have (or need) a snapshot. Preview issues left unfixed before they are pushed to the repository will typically undergo the server-based analysis and commit process. Therefore, these issues will receive a snapshot in Coverity Connect *after* they are initially reported. Whether fixed or left unfixed prior to the push to the source code repository, issues will be identified as Preview issues if they were initially reported through a preview process.

Component	Type	Description
firstDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the First Detected date of a CID. Example1:2013-03-18T12:42:19.384-07:00 Example2:2013-03-
firstDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the First Detected date of a CID. For an example, see firstDetectedEndDate.
fixTargetNameList	string	Fix target for the CID; a triage value for the CID. Multiple fix targets allowed.
impactNameList	string	Probable impact (High, Medium, or Low) of the issue found by the checker. Zero or more impact levels allowed.
issueKindList	string	Issue kind. Multiple issue kinds allowed.
lastDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.
lastDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.
legacyNameList	string	Legacy designation for the CID (true or false), a triage value for the CID. Built-in attribute. Defaults to false.
ownerNameList	string	Owner of the CID.
ownerNamePattern	string	Glob pattern matching the first or last name of the owner of a CID.
severityNameList	string	Severity of the CID; a triage value for the CID. Multiple severities allowed.

#### 4.6.30. Complex Type: projectTrendRecordFilterSpecDataObj

##### 4.6.30.1. Description

Filter for project trend records to return.

##### 4.6.30.2. Derived By

Restricting anyType

##### 4.6.30.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
endDate	dateTime	End date (and optionally, time) for the set of records to retrieve.
startDate	dateTime	Start date (and optionally, time) for the set of records to retrieve.

#### 4.6.31. Complex Type: propertyDataObj

##### 4.6.31.1. Description

A key/value pair for a property of an instance of a software issue.

##### 4.6.31.2. Derived By

Restricting anyType

##### 4.6.31.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
key	string	Identifier for the property.
value	string	Value of the property.

#### 4.6.32. Complex Type: propertySpecDataObj

##### 4.6.32.1. Description

A key/value pair for a property of an instance of a software issue.

##### 4.6.32.2. Derived By

Restricting anyType

##### 4.6.32.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
key	string	Do not use this field. The API does not process these values.
value	string	Do not use this field. The API does not process.

### 4.6.33. Complex Type: snapshotScopeDefectFilterSpecDataObj

#### 4.6.33.1. Description

Filter on the snapshots to return.

#### 4.6.33.2. Derived By

Restricting anyType

#### 4.6.33.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
actionNameList	string	Name/value pairs for a list of attributes.
attributeDefinitionValueFilterMap	Map	Map of an attribute value.
checkerCategoryList	string	List of checker categories.
checkerList	string	List of checkers.
checkerTypeList	string	List of checker types.
cidList	long	List of CIDs.
classificationNameList	string	Classification of the CID. Multiple classifications allowed.
componentIdExclude	boolean	If one or more component name filters is specified, set to true to exclude matching results from the specified components. Defaults to false, including the matches from the components in the results.
componentIdList	componentIdDataObj	Name of a component that contains the CID. Multiple components allowed.
cweList	long	Common Weakness Enumeration identifier of the type of issue found by the checker. Zero or more identifiers allowed.
externalReference	string	An external reference for a CID that is used by your company to identify the software issue. Corresponds to a field in the Coverity Connect triage pane.
fileName	string	A file name. Example: /test.c
firstDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the First Detected date of a CID. Example1: 2013-03-18T12:42:19.384-07:00 Example2: 2013-03-18

Component	Type	Description
firstDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the First Detected date of a CID. For an example, see firstDetectedEndDate.
fixTargetNameList	string	Fix target for the CID; a triage value for the CID. Multiple fix targets allowed.
functionMergeName	string	Internal function name used as one of the criteria for merging separate occurrences of the same software issue, with the result that they are identified by the same CID.
functionName	string	Name of the function or method.
impactNameList	string	Probable impact (High, Medium, or Low) of the issue found by the checker. Zero or more impact levels allowed.
issueComparison	string	If set to <i>PRESENT</i> , returns overlapping CIDs in a snapshot comparison, that is, CIDs found in snapshot(s) to which both the <i>showSelector</i> and <i>compareSelector</i> values of the <i>snapshotScope</i> parameter ( <i>snapshotScopeSpecDataObj</i> ) apply. If set to <i>ABSENT</i> , returns CIDs that are present in the snapshot(s) to which the <i>showSelector</i> value applies but absent from those to which the <i>compareSelector</i> value applies. If not set, values are <i>PRESENT</i> and <i>ABSENT</i> .
issueKindList	string	Issue kind. Multiple issue kinds allowed.
lastDetectedEndDate	dateTime	Ending date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.
lastDetectedStartDate	dateTime	Starting date (and optionally, time) for the date range matching the Last Detected date of a CID. For an example, see firstDetectedEndDate.
legacyNameList	string	Legacy designation for the CID (true or false), a triage value for the CID. Built-in attribute. Defaults to false.
maxOccurrenceCount	int	Maximum number of instances of software issues associated with a given CID. See <i>minOccurrenceCount</i> .
mergeExtra	string	Internal property used as one of the criteria for merging occurrences of an issue.
mergeKey	string	Internal signature used to merge separate occurrences of the same software issue and identify them all by the same CID.

Component	Type	Description
minOccurrenceCount	int	Minimum number of instances of software issues associated with a given CID. See maxOccurrenceCount.
ownerNameList	string	Owner of the CID.
ownerNamePattern	string	Glob pattern matching the first or last name of the owner of a CID.
severityNameList	string	Severity of the CID; a triage value for the CID. Multiple severities allowed.
statusNameList	string	Status of the CID. Multiple statuses allowed.
streamExcludeNameList	streamIdDataObj	Identifier for a stream to exclude. Multiple streams are allowed. See streamExcludeQualifier.
streamExcludeQualifier	string	If set to <i>ANY</i> , the filter will exclude from the results CIDs found in each of the streams listed in the <i>streamExcludeNameList</i> field. If set to <i>ALL</i> , the filter will only exclude a CID if it is found in all listed streams. Valid values are <i>ANY</i> or <i>ALL</i> . Defaults to <i>ANY</i> .
streamIncludeNameList	streamIdDataObj	Identifier for a stream to include. Multiple streams are allowed. See <i>streamIncludeQualifier</i> .
streamIncludeQualifier	string	If set to <i>ANY</i> , the filter will return CIDs found in each of the streams listed in the <i>streamIncludeNameList</i> field. If set to <i>ALL</i> , the filter will only return a CID if it is found in all listed streams. Valid values are <i>ANY</i> or <i>ALL</i> . Defaults to <i>ANY</i> .

#### 4.6.34. Complex Type: snapshotScopeSpecDataObj

##### 4.6.34.1. Description

Specification used for snapshot comparison.

##### 4.6.34.2. Derived By

Restricting anyType

##### 4.6.34.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		

Component	Type	Description
compareOutdatedStreams	boolean	If set to <i>true</i> , includes outdated streams found in snapshots specified by <i>compareSelector</i> . If <i>false</i> , the default, only non-outdated streams are included.  See the note in <i>showOutdatedStreams</i> .
compareSelector	string	Snapshot ID or snapshot grammar value that is used to set the scope of snapshots to compare with the <i>showSelector</i> snapshot scope. For more information, see "Snapshot comparison" in the Coverity Connect Administration Guide.
showOutdatedStreams	boolean	If set to <i>true</i> , includes outdated streams found in snapshots specified by <i>showSelector</i> . If <i>false</i> , the default, only non-outdated streams are included.  Note that a user with proper RBAC permissions at the stream level can designate a stream as <i>outdated</i> to exclude the stream from Coverity Connect processes.
showSelector	string	Require Snapshot ID or snapshot grammar value that is used to set the scope of snapshots  Default: <i>last()</i> which includes the latest snapshot of each stream in the project.  See the Coverity Platform Administration Guide for details on the snapshot grammar.

#### 4.6.35. Complex Type: standardAttributeIdDataObj

##### 4.6.35.1. Description

Identifier for a standard attribute.

##### 4.6.35.2. Derived By

Restricting anyType

##### 4.6.35.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
<b>SEQUENCE</b>		
name	string	Name of the standard attribute.

Component	Type	Description
-----------	------	-------------

#### 4.6.36. Complex Type: standardAttributeValueFilterMapDataObj

##### 4.6.36.1. Description

Filter for one or more standard attribute values.

##### 4.6.36.2. Derived By

Restricting anyType

##### 4.6.36.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
standardAttributeId	standardAttributeIdDataObj	Identifier for the standard attribute to filter.
standardAttributeValues	standardAttributeValuesDataObj	Value of the standard attribute to filter. Multiple values allowed.

#### 4.6.37. Complex Type: standardAttributeValuesDataObj

##### 4.6.37.1. Description

Identifier for a standard attribute value.

##### 4.6.37.2. Derived By

Restricting anyType

##### 4.6.37.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Name of the standard attribute value.

#### 4.6.38. Complex Type: streamDefectDataObj

##### 4.6.38.1. Description

Returns data on a CID within the context of a stream.

#### 4.6.38.2. Derived By

Restricting anyType

#### 4.6.38.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
checkerName	string	Name of the checker that found the issue.
cid	long	CID of the software issue.
defectInstances	defectInstanceDataObj	Data on an instance of a software issue.
defectStateAttributeValues	defectStateAttributeValueDataObj	Triage attribute/value pair for a software issue.
domain	string	Domain of the issue.
history	defectStateDataObj	Historical triage data on an instance of a software issue.
id	streamDefectIdDataObj	Identifier for the software issue within the scope of a stream.
streamId	streamIdDataObj	Identifier for the stream in which the issue occurs.

#### 4.6.39. Complex Type: streamDefectFilterSpecDataObj

##### 4.6.39.1. Description

Filter used to return matching software issues within the scope of one or more streams.

##### 4.6.39.2. Derived By

Restricting anyType

##### 4.6.39.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
defectStateEndDate	dateTime	Ending date (and optionally, time) for the CIDs to return.
defectStateStartDate	dateTime	Starting date (and optionally, time) for the CIDs to return.
includeDefectInstances	boolean	Set to <i>true</i> for data on each instance of software issue, including the ID. Defaults to <i>false</i> .

Component	Type	Description
includeHistory	boolean	Set to <i>true</i> for historical triage data on each instance of the software issue.
streamIdList	streamIdDataObj	Identifier for a stream. Multiple streams allowed.

#### 4.6.40. Complex Type: streamDefectIdDataObj

##### 4.6.40.1. Description

Identifier for a software issue within the scope of a stream.

##### 4.6.40.2. Derived By

Restricting anyType

##### 4.6.40.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
 SEQUENCE		
defectTriageld	long	Internal value for the last known triage ID. This ID changes when developers triage the issue that is associated with the <i>id</i> .
defectTriageVerNum	int	Internal value for the last known triage version. This number changes when developers triage the issue that is associated with the <i>id</i> .
id	long	Internal identifier for the software issue within the context of the stream.
verNum	int	Version number associated with the <i>id</i> .

#### 4.6.41. Complex Type: streamIdDataObj

##### 4.6.41.1. Description

Identifier for the stream.

##### 4.6.41.2. Derived By

Restricting anyType

##### 4.6.41.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
name	string	Required. Name of the stream. You can specify one or more instances of streamIdDataObj. See the example.

#### 4.6.42. Complex Type: triageHistoryDataObj

##### 4.6.42.1. Description

Data object that capture state of triage attributes for a CID at a given point in time. For example, if a developer designates a new CID as a bug, the triage state changes.

##### 4.6.42.2. Derived By

Restricting anyType

##### 4.6.42.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		
attributes	defectStateAttributeValueDataObj	Triage attributes and values.
id	long	Identifier that groups a set of triage changes.

#### 4.6.43. Complex Type: triageStoreIdDataObj

##### 4.6.43.1. Description

Identifier for the triage store.

##### 4.6.43.2. Derived By

Restricting anyType

##### 4.6.43.3. Content Model

Contains elements as defined in the following table.

Component	Type	Description
SEQUENCE		

Component	Type	Description
name	string	Required. Name of the triage store.

## 4.7. Types and Constants

### 4.7.1.

Enumerations and string-based values that are valid for specified fields and parameters.

### 4.7.2. Action

#### 4.7.2.1.

Action to take on CID	Description
Undecided	Default for an Unclassified issue. Built-in Action attribute value that a user can set.
Fix Required	CID that needs to be fixed. Built-in Action attribute value that a user can set.
Fix Submitted	CID for which a fix has been submitted for analysis. Built-in Action attribute value that a user can set.
Modeling Request	CID for which a user has requested function modelling of the underlying source code. Built-in Action attribute value that a user can set.
Ignore	CID that can be ignored. Built-in Action attribute value that a user can set.
[custom_action]	Action created through a custom Action attribute value.

### 4.7.3. Attribute Name

#### 4.7.3.1.

attributeDefinitionId.name	Description
Action	The built-in Action attribute used to triage a CID.
Classification	The built-in Classification attribute used to triage a CID. Values: Unclassified, Pending, False Positive, Intentional, Bug.
Comment	The built-in Comment attribute used to add a comment about a CID.
DefectStatus	The built-in Status attribute used to triage a CID.
ExternalReference	The built-in Ext. Reference attribute.
Fix Target	The built-in Fix Target attribute.
Legacy	The built-in Legacy attribute.
Owner	The built-in Owner attribute.
OwnerName	The first and/or last name of the Owner.
Severity	The built-in Severity attribute used to triage a CID.

**attributeDefinitionId.name**    **Description**

---

TranslatedOwner	The translated username of the Owner.
[custom_action]	A custom attribute used to triage a CID.

---

**4.7.4. Attribute Type (attributeType)****4.7.4.1.****Attribute Type**                      **Description**

---

LIST_OF_VALUES	Type for the pick list of specified attribute values.
STRING	Type for text-based attribute values.

---

**4.7.5. Date Time (dateTime)****4.7.5.1.**

Valid SOAP formats for this type can include the date and time, or omit the time. When specifying the dateTime, you will need to use a format that is supported by your SOAP client library. The following come from SOAP request and response examples in this guide.

*Example1:*2013-03-19

*Example2:*

2013-03-19T14:40

*Example3:*2013-03-19T14:40:32.022-07:00

- Date = 2013-03-19
- Time = T14:40:32.022
- Time Zone (UTC) = -07:00

**4.7.6. Domain****4.7.6.1.****Checker Domain**                      **Description**

---

STATIC_C	Applies to checkers that run static analyses of C/C++ source code.
STATIC_JAVA	Applies to checkers that run static analyses of Java source code.
STATIC_CS	Applies to checkers that run static analyses of C# source code.
DYNAMIC_JAVA	Applies to checkers that run dynamic analyses of Java source code. Coverity Dynamic Analysis checkers.
EXTEND	Applies to custom checkers created through the Coverity Extend SDK.

---

## 4.7.7. Event Kind (eventKind)

### 4.7.7.1.

Event Kind	Description
MODEL	Corresponds to a function call. In Coverity Connect, model events appear beside a <i>Show Details</i> link.
PATH	Identifies a conditional branch and the decision necessary for the software issue to occur.  Example:Condition !p, taking false branch  Related lines 107-108 of sample code: 107 if (!p) 108 return NO_MEM;
MULTI	Provides evidence from the source code that supports the checker's finding of a software issue. Also called an Evidence event.
NORMAL	References a line of code that is identified as a contributing factor to the software issue found by the checker.  Examples:  1. alloc_fn: Storage is returned from allocation function malloc.  2. var_assign: Assigning: p = storage returned from malloc(12U)  Related line 5 of sample code: 5 char *p = malloc(12);
REMIEDIATION	Provides remediation advice that is intended to help you fix the reported software issue, rather than report what is wrong. Used in security defects.

## 4.7.8. Issue Kind (issueKind)

### 4.7.8.1.

Issue Kind	Description
QUALITY	Quality issue found by Coverity checkers.
SECURITY	Potential security vulnerability found by Coverity checkers.
TEST	Policy violations of developer tests found by Test Advisor.

## 4.7.9. Language

### 4.7.9.1.

Programming Language	Description
MIXED	The stream can contain results from any programming language. This is the recommended and default setting.

Programming Language	Description
	Other values exist only for backward compatibility.
	Please note: In the Coverity Connect "Projects & Streams" Configuration dialog, the "Any" element of the Language drop-down is equivalent to "MIXED" in the Web Services API.
CSHARP	C#
CXX	C and/or C++
JAVA	Java
OTHER	Languages other than C, C++, C#, and Java.

#### 4.7.10. Locale

##### 4.7.10.1.

Locale of the User	Description
en_US	US English
ja_JP	Japanese
zh_CN	Simplified Chinese

#### 4.7.11. Permission

##### 4.7.11.1.

Permissions	Description	Type
accessWebUI	Log in.	global
accessWS	Access Web services.	global
commitToStream	Commit analysis results (software issues and related resources) to a stream.	streams
createProjects	Create projects.	projects
createStreams	Create streams.	projects
createTriageStores	Create triage stores.	global
manageAttributes	Manage attributes.	global
manageComponentMaps	Manage component maps.	global
manageHierarchies	Manage Policy Manager Hierarchies: Administrative access.	global
manageProjects	Manage projects.	projects
manageRoleDefinitions	Manage role definitions.	global

Permissions	Description	Type
manageServerParams	Manage server parameters.	global
manageStreams	Manage streams.	streams
manageTriageStores	Manage triage stores.	triage store
manageUsersGroups	Manage user groups.	global
previewCommit	Preview commits to streams.	streams
triageDefects	Triage CIDs.	streams, component, triage store
viewDefects	View software issues.	streams, component, triage store
viewIntegrityControl	View Policy Manager (previously called "Integrity Control"): End user access.	global
viewGlobalDashboard	Access the dashboard (charts and graphs) for the current project for Coverity and Test Advisor results.	global
viewProjectsHistoryAndTrends	View project trends and reports.	projects
viewSource	View source code in the UI.	streams, component, triage store

#### 4.7.12. Role Assignment Type (roleAssignmentType)

##### 4.7.12.1.

Role Assignment Type	Description
group	Role that is assigned to a group.
user	Role that is assigned to a user.

#### 4.7.13. Role Type (roleAssignmentDataObj.type)

##### 4.7.13.1.

Role Type	Description
component	For applying specified roleId at the component level. Example: The built-in Developer role.
global	For applying specified roleId at the top (global) level. Example: The Roles assigned when creating or updating users or groups take place at this level.
project	For applying specified roleId at the project level.
stream	For applying specified roleId at the stream level.
triageStore	For applying specified roleId at the triage store level.

#### 4.7.14. Role ID (Built-in Roles: roleAssignmentDataObj.roleId)

##### 4.7.14.1.

Attribute Type	Coverity Connect Role	Description
committer	Committer	Editable and deletable. Supports committing analysis results to the database. Default permissions: commitToStream
desktopDeveloper	Desktop Developer	Editable and deletable. Supports permissions commonly needed by Coverity Desktop developers. Default permissions: triageDefects, accessWS, viewDefects, createStreams, manageProjects, viewSource, accessWebUI
developer	Developer	Editable and deletable. Supports permissions commonly needed by developers who use Coverity Connect. Default permissions: triageDefects, accessWS, viewDefects, viewSource, accessWebUI, viewProjectsHistoryAndTrends
hierarchyAdmin	Hierarchy Administrator	Not editable or deletable. Supports the creation of Policy Manager Hierarchies. Default permissions: accessWS, manageHierarchies, viewIntegrityControl, accessWebUI
icUser	Policy Manager User	Not editable or deletable. Supports end usage of Policy Manager, which includes creating, editing, and using charts and heatmaps. Default permissions: accessWS, accessWebUI, viewIntegrityControl
noAccess	No Access	Not editable or deletable. Prohibits access. No permissions are enabled.
observer	Observer	Editable and deletable. Default permissions: accessWS, viewDefects, viewSource, accessWebUI, viewProjectsHistoryAndTrends
projectAdmin	Project Admin	Editable and deletable. Supports permissions that are commonly needed by project administrators. Default permissions: accessWS, manageComponentMaps, createTriageStores, manageAttributes, createProjects, accessWebUI
projectOwner	Project Owner	Not editable or deletable. Supports permissions needed by project owners. Default permissions: triageDefects, commitToStream, accessWS, viewDefects, createStreams, manageProjects, viewSource, accessWebUI, manageStreams, viewProjectsHistoryAndTrends
reporter	System Report Generator	Not editable or deletable. Default permissions: viewDefects, viewSource
serverAdmin	Server Admin	Editable and deletable. Default permissions: accessWS, accessWebUI, manageRoleDefinitions, manageUsersGroups, manageServerParams

Attribute Type	Coverity Connect Role	Description
streamAdmin	Stream Admin	Editable and deletable. Default permissions: accessWS, createStreams, accessWebUI
streamOwner	Stream Owner	Not editable or deletable. Default permissions: triageDefects, commitToStream, accessWS, viewDefects, viewSource, accessWebUI, manageStreams
sysAdmin	System Admin	Not editable or deletable. Default permissions: <i>All permissions</i>
triageStoreOwner	Triage Store Owner	Not editable or deletable. Default permissions: accessWS, viewDefects, triageDefects, manageTriageStores, accessWebUI
visitor	Visitor	Not editable or deletable. . Default permissions: accessWS, accessWebUI
wsReporter	Web Service Reporter	Editable and deletable. Default permissions: accessWS, viewDefects

#### 4.7.15. Severity

##### 4.7.15.1.

Severity of a CID	Description
Unspecified	Default for unclassified CIDs. Built-in but editable Severity attribute value that a user can set.
Major	Designation for a major software issue. Built-in but editable Severity attribute value that a user can set.
Minor	Designation for a minor software issue. Built-in but editable and deletable Severity attribute value that a user can set.
Moderate	Designation for a moderate software issue. Built-in but editable and deletable Severity attribute value that a user can set.
[custom_severity]	Severity created through a custom Severity attribute value.

#### 4.7.16. Status

##### 4.7.16.1.

Status of CID	Description
New	Automatically applied to issues that are classified as <i>Unclassified</i> . Built-in Status attribute value.
Triaged	Automatically applied to issues that are classified by a user as <i>Pending</i> or <i>Bug</i> . Built-in Status attribute value.
Dismissed	Automatically applied to issues that are classified by a user as <i>Intentional</i> or <i>False Positive</i> . Built-in Status attribute value.

<b>Status of CID</b>	<b>Description</b>
Absent Dismissed	Automatically applied to issues that have been classified as <i>Intentional</i> or <i>False Positive</i> in an earlier snapshot, but are absent from the latest snapshot. Built-in Status attribute value.
Fixed	Automatically applied to issues that occurred in an earlier snapshot of the analysis results but do not appear in a later snapshot (either the current or specified snapshot). Built-in Status attribute value.

---

---

## Chapter 5. View Management API

### Table of Contents

5.1. view/v1/username/{source_user}/ldap/{source_ldap} .....	187
5.2. view/v1/view/{view_id} .....	189
5.3. view/v1/views .....	191
5.4. views/v1 .....	194
5.5. viewContents/{type}/v1/{view_id} .....	197

The View Management API specifies a set of identifiable resources and URI formats that you can use to interact with Coverity Connect views. This API allows you to use standard HTTP to retrieve resources in JSON (and in some cases, CSV) format.

See the individual operation descriptions for more information on formatting your request and about the expected output.

### 5.1. PATCH `/api/view/v1/username/{source_user}/ldap/{source_ldap}`

#### 5.1.1. Description

**HTTP Method:** PATCH

**URI format:** `/api/view/v1/username/{source_user}/ldap/{source_ldap}`

Reassign ownership of all views owned by one user (the source user) to a different user (the target user). Execution requires the **System Admin** role on all affected projects; otherwise returns an error. Returns an error if the source and target users are the same user.

Following the execution of this operation, the source user will not own any views. The next time the source user logs in, Coverity Connect will create a new set of default views for the source user.

#### 5.1.2. Parameters

##### 5.1.2.1. Path Parameters

Parameter name	Description
source_user	<b>Required.</b> The username of the current owner of the views whose ownership you want to reassign (the source user).
source_ldap	<b>Required.</b> Address of the LDAP server that authenticates the source user.

##### 5.1.2.2. Payload Parameters

Payload parameters are specified in the HTTP header using `--data`.

Parameter name	Description
target_username	<b>Required.</b> The username of the new owner of the views whose ownership you want to reassign (the target user).
target_ldap	<b>Required.</b> Address of the LDAP server that authenticates the target user.

### 5.1.3. Response

The response body contains a JSON array defined by the following name-value pairs.

Name	Value	JSONPath
Updated Views	<b>Type:</b> array  An array whose elements represent the views whose ownership has been reassigned.	<code>\$.["Updated Views"]</code>
owner	<b>Type:</b> object  Identifies the target user (the user to whom ownership of the view has been reassigned).	<code>\$.["Updated Views"] [*].owner</code>
username	<b>Type:</b> string  Login name of the target user (the user to whom ownership of the view has been reassigned).	<code>\$.["Updated Views"] [*].owner.username</code>
ldapServer	<b>Type:</b> string  Address of the LDAP server that authenticates the target user (the user to whom ownership of the view has been reassigned).	<code>\$.["Updated Views"] [*].owner.ldapServer</code>
viewkey	<b>Type:</b> string  Identifies the table that contains the reassigned view.	<code>\$.["Updated Views"] [*].viewkey</code>
name	<b>Type:</b> string  Name of the reassigned view.  Except in the case of default views, the name is a concatenation of the original name, an underscore, and the source user name, for example: "Original Table Name_my-source-user-name". Default views are not renamed.	<code>\$.["Updated Views"] [*].name</code>

## 5.1.4. Examples

### Example request for JSON output

```
curl -G -X PATCH --data
"target_username=my_target_user&target_ldap=my_target_ldap_server.my_domain.com"
--user "admin:pass" "http://localhost:8080/api/view/v1/username/
my_source_user/ldap/my_source_ldap_server.my_domain.com"
```

### Example JSON response body

```
{
  "Updated Views": [
    {
      "owner": {
        "username": "my_target_user",
        "ldapServer": "my_target_ldap_server.my_domain.com"
      },
      "viewkey": "HierarchiesTable",
      "name": "All Hierarchies"
    },
    {
      "owner": {
        "username": "my_target_user",
        "ldapServer": "my_target_ldap_server.my_domain.com"
      },
      "viewkey": "FilesTable",
      "name": "Uncovered By Tests_my_source_user"
    }
  ]
}
```

## 5.2. PATCH api/view/v1/view/{view\_id}

### 5.2.1. Description

**HTTP Method:** PATCH

**URI format:** /api/view/v1/view/{view\_id}

Reassign ownership of the specified view from the current owner (the source user) to the specified user (the target user). Execution requires the **System Admin** role on the affected project; otherwise returns an error. Returns an error if the source and target users are the same user.

Following the execution of this operation, the source user will own fewer views. The next time the source user logs in, Coverity Connect will create new default views for any that are missing from the source user's set of default views.

## 5.2.2. Parameters

### 5.2.2.1. Path Parameters

Parameter name	Description
view_id	<b>Required.</b> Name or numeric ID of the view whose ownership you want to reassign.

### 5.2.2.2. Payload Parameters

Payload parameters are specified in the HTTP header using `--data`.

Parameter name	Description
username	<b>Required.</b> The user to whom you want to assign ownership of the view (the target user).
ldap_server	<b>Required.</b> Address of the LDAP server that authenticates the target user.

## 5.2.3. Response

The response body contains a JSON array defined by the following name-value pairs.

Name	Value	JSONPath
Updated View	<b>Type:</b> object  An object that represents the view whose ownership has been reassigned.	<code>\$.["Updated View"]</code>
owner	<b>Type:</b> object  Identifies the target user (the user to whom ownership of the view has been reassigned).	<code>\$.["Updated View"].owner</code>
username	<b>Type:</b> string  Login name of the target user (the user to whom ownership of the view has been reassigned).	<code>\$.["Updated View"].owner.username</code>
ldapServer	<b>Type:</b> string  Address of the LDAP server that authenticates the target user (the user to whom ownership of the view has been reassigned).	<code>\$.["Updated View"].owner.ldapServer</code>
viewkey	<b>Type:</b> string	<code>\$.["Updated View"].viewkey</code>

Name	Value	JSONPath
name	<p>Identifies the table that contains the reassigned view.</p> <p><b>Type:</b> string</p> <p>Name of the reassigned view.</p> <p>Except in the case of default views, the name is a concatenation of the original name, an underscore, and the source user name, for example: "Original Table Name_my-source-user-name". Default views are not renamed.</p>	\$.["Updated View"].name

## 5.2.4. Examples

### Example request for JSON output

```
curl -G -X PATCH --data
"username=my_target_user&ldap_server=my_target_ldap_server.my_domain.com" --
user "admin:pass" "http://localhost:8080/api/view/v1/view/10030"
```

### Example JSON response body

```
{
  "Updated View": {
    "owner": {
      "username": "my_target_user",
      "ldapServer": "my_target_ldap_server.my_domain.com"
    },
    "viewkey": "FilesTable",
    "name": "Uncovered By Tests_my_target_user"
  }
}
```

## 5.3. GET api/view/v1/views

### 5.3.1. Description

**HTTP Method:** GET

**URI format:** /api/view/v1/views

Retrieve all views. Execution requires the **System Admin** role on all projects; otherwise returns an error.

### 5.3.2. Parameters

None

### 5.3.3. Response

The response body contains a JSON array defined by the following name-value pairs.

Name	Value	JSONPath
All Views Status	<b>Type:</b> array  An array whose elements represent the views visible to the current user.	\$.["All Views Status"]
viewId	<b>Type:</b> number  Identifier for the view.	\$.["All Views Status"] [*].viewId
viewName	<b>Type:</b> string  Name of the view.	\$.["All Views Status"] [*].viewName
owner	<b>Type:</b> object  Identifies the user who owns the view.	\$.["All Views Status"] [*].owner
username	<b>Type:</b> string  Login name of the user who owns the view.	\$.["All Views Status"] [*].owner.username
ldapServer	<b>Type:</b> string  LDAP server that authenticates the user who owns the view.	\$.["All Views Status"] [*].owner.ldapServer
sharedUsers	<b>Type:</b> array  Identifies users who share the view. The array is empty if no users share the view.	\$.["All Views Status"] [*].sharedUsers
sharedGroups	<b>Type:</b> array  Identifies groups that share the view. The array is empty if no groups share the view.	\$.["All Views Status"] [*].sharedGroups
hasViewNotifications	<b>Type:</b> boolean  Indicates whether the view has notifications.  <ul style="list-style-type: none"> <li>• <code>true</code> – the view has notifications.</li> <li>• <code>false</code> – the view does not have notifications.</li> </ul>	\$.["All Views Status"] [*].hasViewNotifications

Name	Value	JSONPath
ownerActive	<p><b>Type:</b> boolean</p> <p>Indicates whether the user who owns the view is active.</p> <ul style="list-style-type: none"> <li>• <code>true</code> – the user who owns the view is active.</li> <li>• <code>false</code> – the user who owns the view is not active.</li> </ul>	<code>\$.["All Views Status"] [*].ownerActive</code>
ownerDeleted	<p><b>Type:</b> boolean</p> <p>Indicates whether the user who owns the view has been deleted.</p> <ul style="list-style-type: none"> <li>• <code>true</code> – the user who owns the view has been deleted.</li> <li>• <code>false</code> – the user who owns the view has not been deleted.</li> </ul>	<code>\$.["All Views Status"] [*].ownerDeleted</code>

### 5.3.4. Examples

#### Example request for JSON output

```
curl --user admin:password "localhost:8080/api/view/v1/views"
```

#### Example request for CSV file

```
curl --header "Accept: text/csv" --user admin:password "localhost:8080/api/view/v1/views" > outputFile.csv
```

#### Example JSON response body

```
{
  "All Views Status": [
    {
      "viewId": "10002",
      "viewName": "Currently Failing",
      "owner": {
        "username": "admin",
        "ldapServer": "my_ldap_server.my_domain.com"
      },
      "sharedUsers": [],
      "sharedGroups": [],
      "hasViewNotifications": false,
      "ownerActive": true,
    }
  ]
}
```

```

    "ownerDeleted":false
  },
  {
    "viewId":"10006",
    "viewName":"Uncovered By Tests",
    "owner": {
      "username":"admin",
      "ldapServer":"my_ldap_server.my_domain.com"
    },
    "sharedUsers":[],
    "sharedGroups":[],
    "hasViewNotifications":false,
    "ownerActive":true,
    "ownerDeleted":false
  }
]
}

```

### Example CSV file

```

curl -H "Accept: text/csv" --user "admin:pass" "http://localhost:8080/api/view/v1/views"
VIEW ID|VIEW NAME|OWNER|SHARED USERS|SHARED GROUPS|ACTIVE OWNER|DELETED OWNER|HAS NOTIFICA
10002,Currently Failing,(admin;my_ldap_server.my_domain.com),,,true,false,false
10006,Uncovered By Tests,(admin;my_ldap_server.my_domain.com),,,true,false,false
10114,High Issue Density (>1),(testing;my_ldap_server.my_domain.com),,,true,false,false

```

## 5.4. GET api/views/v1

### 5.4.1. Description

**HTTP Method:** GET

**URI format:** /api/views/v1/

Retrieve all views accessible to the specified user.

#### Note

The user must have logged into Coverity Connect at least once before requesting this resource. Otherwise, the request results in an error.

### 5.4.2. Parameters

None

### 5.4.3. Response

The response body contains a JSON array defined by the following name-value pairs.

---

Name	Value	JSONPath
views	<b>Type:</b> array  An array whose elements represent the views visible to the current user.	<code>\$.views</code>
id	<b>Type:</b> number  Identifier for the view.	<code>\$.views[*].id</code>
type	<b>Type:</b> string  The type of view. Describes the type information the view displays in Coverity Connect.	<code>\$.views[*].type</code>
name	<b>Type:</b> string  Name of the view.	<code>\$.views[*].name</code>
groupBy	<b>Type:</b> boolean  Indicates whether the view's <b>Group By</b> option is set. <ul style="list-style-type: none"><li>• <code>true</code> – the view's <b>Group By</b> option is set.</li><li>• <code>false</code> – the view's <b>Group By</b> option is not set.</li></ul> For information about the <b>Group By</b> view option, see the section "Group By" in the <i>Coverity Platform 2020.12 User and Administrator Guide</i> .	<code>\$.views[*].groupBy</code>
columns	<b>Type:</b> array  An array whose elements describe the columns included in the view. Each array element is an object with the following name-value pairs: <ul style="list-style-type: none"><li>• <code>name</code> – the column's identifier.</li><li>• <code>label</code> – the column's user-visible label.</li></ul>	<code>\$.views[*].columns</code>

---

#### 5.4.4. Examples

##### Example request

```
curl --user admin:password "localhost:8080/api/views/v1"
```

**Example JSON response body**

```
{
  "views": [
    {
      "id": 12345,
      "type": "files",
      "name": "All Files",
      "groupBy": false,
      "columns": [
        {
          "name": "cid",
          "label": "CID"
        },
        {
          "name": "displayType",
          "label": "Type"
        }
      ]
    },
    {
      "id": 54321,
      "type": "issuesByProject",
      "name": "All In Project",
      "groupBy": false,
      "columns": [
        {
          "name": "cid",
          "label": "CID"
        },
        {
          "name": "displayType",
          "label": "Type"
        },
        {
          "name": "displayImpact",
          "label": "Impact"
        },
        {
          "name": "firstDetected",
          "label": "First Detected"
        },
        {
          "name": "owner",
          "label": "Owner"
        }
      ]
    }
  ]
}
```

```
]
}
```

## 5.5. GET api/viewContents/{type}/v1/{view\_id}

### 5.5.1. Description

**HTTP Method:** GET

**URI format:** /api/viewContents/{type}/v1/{view\_id}?projectId={string or num}  
[&rowCount={num}][&offset={num}][&sortKey={string}][&sortOrder=[asc|desc]]

Retrieve the contents of the specified table view in Coverity Connect.

### 5.5.2. Parameters

#### 5.5.2.1. Path Parameters

Parameter name	Description
type	<p><b>Required.</b> The type of view requested. Possible values:</p> <ul style="list-style-type: none"> <li>issues</li> <li>issuesByProject</li> <li>functions</li> <li>files</li> <li>components</li> <li>checkers</li> <li>owners</li> <li>projects</li> <li>tests</li> <li>trends</li> <li>snapshots</li> </ul> <p>The type <code>projects</code> can be used to export views at the <b>Projects (View all)</b> level, such as the <b>All Projects</b> view. All other view types apply to views at the single-project level, such as the <b>Outstanding Defects</b> view. For example, the following retrieves all rows (all projects) of the <b>All Projects</b> view:</p> <pre>localhost:8080/api/viewContents/projects/v1/All%20Projects?projectId=*</pre> <p>In contrast, this example retrieves only one row (the project named <code>my_project</code>):</p> <pre>localhost:8080/api/viewContents/projects/v1/All%20Projects?projectId=my_project</pre>
view_id	<p><b>Required.</b> Name or numeric ID of the view.</p>

### 5.5.2.2. Query Parameters

Parameter name	Description
projectId	<b>Required.</b> The numeric or string name identifier of the Coverity Connect project with which the view is associated.
rowCount	Maximum number of rows to return in this query. The default value is 100. To return all rows, set the value to -1.
offset	Offset to use when accessing rows; this is the number of rows that will be skipped from the top of the sorted results. The default value is 0.
sortKey	Name of the column on which to sort the view. If unspecified, Coverity Connect chooses based on context.
sortOrder	Specifies the table's sort direction. Possible values: <ul style="list-style-type: none"><li>• <code>asc</code> – ascending</li><li>• <code>desc</code> – descending (Default)</li></ul>

---

### 5.5.3. Response

The response body contains a JSON object defined by the following name-value pairs.

Name	Value	JSONPath
viewContentsV1	<b>Type:</b> object  An object representing the view contents.	<code>\$.viewContentsV1</code>
offset	<b>Type:</b> number  The number of rows that were skipped from the top of the sorted results, if any.	<code>\$.viewContentsV1.offset</code>
totalRows	<b>Type:</b> number  The total number of rows in the table view, as it exists on Coverity Connect. Depending on the value specified in the <code>rowCount</code> query parameter, this value might not match the number of rows in the output.	<code>\$.viewContentsV1.totalRows</code>
columns	<b>Type:</b> array  An array whose elements describe the columns included in the view. Each array element is an object with the following name-value pairs:	<code>\$.viewContentsV1.columns</code>

---

Name	Value	JSONPath
	<ul style="list-style-type: none"> <li>• <code>name</code> – the column's identifier.</li> <li>• <code>label</code> – the column's user-visible label.</li> </ul>	
<code>rows</code>	<p><b>Type:</b> array</p> <p>An array whose elements each describe one row in the view. Each array element is an object with a set of name-value pairs, each of which identifies a column (using the column's <code>name</code> value as specified in the <code>columns</code> array) and provides a corresponding value, for example, <code>"cid": 12345</code>. As such, the set of name-value pairs in each <code>rows</code> array element correlates with the set of elements in the <code>columns</code> array and varies from view to view.</p>	<code>\$.viewContentsV1.rows</code>

## 5.5.4. Examples

### Example request for JSON output

```
curl --user admin:password "localhost:8080/api/viewContents/issues/v1/Outstanding%20Defects?projectId=sample-app&rowCount=25"
```

### Example request for CSV file

```
curl --header "Accept: text/csv" --user admin:password "localhost:8080/api/viewContents/issues/v1/Outstanding%20Defects?projectId=sample-app&rowCount=25" > outputFile.csv
```

### Example JSON response body

```
{
  "viewContentsV1": {
    "offset": 0,
    "totalRows": 2,
    "columns": [
      {
        "name": "cid",
        "label": "CID"
      },
      {
        "name": "displayType",
        "label": "Type"
      }
    ]
  }
}
```

```
{
  {
    "name": "displayImpact",
    "label": "Impact"
  },
  {
    "name": "status",
    "label": "Status"
  },
  {
    "name": "firstDetected",
    "label": "First Detected"
  },
  {
    "name": "owner",
    "label": "Owner"
  },
  {
    "name": "classification",
    "label": "Classification"
  },
  {
    "name": "severity",
    "label": "Severity"
  }
},
"rows": [
  {
    "cid": 12345,
    "displayType": "Insufficient function coverage",
    "displayImpact": "Low",
    "status": "New",
    "firstDetected": "05/17/14",
    "owner": "Unassigned",
    "classification": "Unclassified",
    "severity": "Unspecified"
  },
  {
    "cid": 54321,
    "displayType": "Insufficient function coverage",
    "displayImpact": "Low",
    "status": "New",
    "firstDetected": "05/17/14",
    "owner": "Unassigned",
    "classification": "Unclassified",
    "severity": "Unspecified"
  }
]
}
```

}

---

## Chapter 6. Error Codes

### Table of Contents

6.1. 1000-1099: General .....	202
6.2. 1100-1199: User .....	203
6.3. 1200-1299: Attribute .....	203
6.4. 1300-1399: Project, stream, snapshot .....	204
6.5. 1400-1499: Defect .....	205
6.6. 1500-1599: Internal .....	205
6.7. 1600-1699: Component .....	205
6.8. 1700-1799: Role .....	206
6.9. 1800-1899: File and function .....	206
6.10. 1900-1999: License .....	206
6.11. 2000-2099: Triage store .....	206
6.12. 2100-2199: LDAP .....	207
6.13. 2200-2299: View Notification .....	207

Descriptions of error codes returned for invalid Web service requests. Codes that were introduced in the latest release appear in bold font face.

#### SOAP Example (Error 1300 to an `updateProject()` request):

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"> <S:Body> <S:Fault
xmlns:ns4="http://www.w3.org/2003/05/soap-envelope"> <faultcode>S:Server</faultcode>
<faultstring>No stream found for name EXAMPLE-cpp.</faultstring> <detail> <ns2:CoverityFault
xmlns:ns2="http://ws.coverity.com/v8"> <errorCode>1300</errorCode> <message>No stream found for
name EXAMPLE-cpp.</message> </ns2:CoverityFault> </detail> </S:Fault> </S:Body></S:Envelope>
```

#### Authentication errors produce a error message without an error code. For example:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"> <S:Body> <S:Fault
xmlns:ns4="http://www.w3.org/2003/05/soap-envelope"> <faultcode>S:Server</faultcode>
<faultstring>User authentication failed.</faultstring> </S:Fault> </S:Body></S:Envelope>
```

### 6.1. 1000-1099: General

#### 6.1.1.

Error Code	Condition
1001	Authentication failed.
1003	Access denied.
1004	Internal error.
1006	Invalid argument passed to service.
1007	Argument limit exceeded.

<b>Error Code</b>	<b>Condition</b>
1008	Argument input exceeded.
1009	Unsupported argument.
1010	Unsupported operation.
<b>1011</b>	A filterSpec field is missing.

---

## **6.2. 1100-1199: User**

### **6.2.1.**

<b>Error Code</b>	<b>Condition</b>
1100	User not found.
1101	Group not found.
1102	Cannot create LDAP user.
1103	Password must not be empty.
1104	Invalid user specification passed to createUser() operation.
1105	Attempt to create user with a duplicate username.
1106	User already deleted.
1107	Group name already in use.
1108	Invalid usage of a built-in group.
1109	Cannot create LDAP group.
1110	LDAP group not found in directory.
1111	Cannot add user to non-local (LDAP) group.
1112	Mail has not been configured.
1113	Unable to notify user.
1114	Mail not configured.
1115	Domain not found.
1116	Cannot change domain of non-local group.
1117	Cannot set domain of non-local group.
1118	Domain not configured.
1119	Multiple domain occurrence.
1120	User deleted failed references.

---

## **6.3. 1200-1299: Attribute**

### **6.3.1.**

<b>Error Code</b>	<b>Condition</b>
1200	Attribute name not found.

---

Error Code	Condition
1207	Attribute list contains duplicates.
1209	Built-in attribute is not deletable.
1210	No custom attributes defined.
1211	Field cannot be modified.
1212	Maximum number of attributes reached.

## 6.4. 1300-1399: Project, stream, snapshot

### 6.4.1.

Error Code	Condition
1300	Stream not found.
1301	Stream exists already.
1302	Project not found.
1303	Project exists already.
1304	Snapshot not found.
1305	Stream type cannot be changed.
1306	Stream is not deletable.
1307	Invalid stream specification.
1308	Invalid project specification.
1309	Invalid project stream list.
1310	Language not supported.
1311	Stream type not valid.
1312	Disposition not valid.
1313	Domain not supported.
1314	Regular expression pattern not valid.
1315	Stream language cannot be changed.
1316	Glob pattern not valid.
1317	Unable to assign a disabled user as defect owner.
1318	Internal stream cannot be managed.
<b>1319</b>	Project needed by hierarchy

## 6.5. 1400-1499: Defect

### 6.5.1.

Error Code	Condition
1400	Defect not found.
1401	Defect edit conflict occurred.
1402	Checker name invalid.
1403	Checker subcategory invalid.
1404	Access to stream defect denied.
1405	Key exists already.
1406	Key required.
1407	Defect attribute limit exceeded.
1408	Defect edit during commit.

---

## 6.6. 1501-1599: Internal errors

### 6.6.1.

Error Code	Condition
1500	Stream locked by another process.
1501	Deleting multiple snapshots from multiple streams.
1502	Snapshot filter too complex.

---

## 6.7. 1600-1699: Component

### 6.7.1.

Error Code	Condition
1600	Component map not found.
1601	Component map not deletable.
1602	Component map not updatable.
1603	More than one default ( <i>Other</i> ) component specified for component map.
1604	No default ( <i>Other</i> ) component specified for component map.
1605	Default component specified for creation.
1606	Invalid component filter.
1607	Component not found.
1608	Component not found.
1609	Default component map not renamable.

---

---

Error Code	Condition
1610	Component map exists already.

---

## 6.8. 1700-1799: Role

### 6.8.1.

---

Error Code	Condition
1700	Role not found.
1701	Permission not found.
1702	Role exists already.
1703	Role not deletable.
1704	Role not editable.
1705	Cannot create built-in role.

---

## 6.9. 1800-1899: File and function

### 6.9.1.

---

Error Code	Condition
1800	Stream file not found.
1801	File contents not found.

---

## 6.10. 1900-1999: License

### 6.10.1.

---

Error Code	Condition
1900	Invalid license.
1901	Invalid license file.
1902	Expired license.
1904	LOC (lines of code) exceeded license limit.
1905	User count exceeds license limit.
1906	Unexpected license error.

---

## 6.11. 2000-2099: Triage store

### 6.11.1.

---

Error Code	Condition
2000	Triage store not found.

---

<b>Error Code</b>	<b>Condition</b>
2001	Triage store already exists.
2002	Triage store not deletable.
2003	Invalid triage store.

---

## **6.12. 2100-2199: LDAP**

### **6.12.1.**

<b>Error Code</b>	<b>Condition</b>
2100	LDAP configuration already exists.
2101	LDAP configuration not found.
2102	LDAP configuration not deletable.
2103	LDAP configuration error.
2104	Unexpected LDAP configuration error.
<b>2105</b>	LDAP not configured.

---

## **6.13. 2200-2299: View Notification**

### **6.13.1.**

<b>Error Code</b>	<b>Condition</b>
2200	View not found.

---

---

## Chapter 7. Glossary

### Table of Contents

7.1. attribute .....	208
7.2. checker .....	208
7.3. CID .....	208
7.4. component .....	209
7.5. defect .....	209
7.6. external reference .....	209
7.7. fix target .....	209
7.8. permission .....	209
7.9. project .....	209
7.10. role .....	210
7.11. snapshot .....	210
7.12. source version .....	210
7.13. stream .....	210
7.14. subcategory .....	210
7.15. target .....	211
7.16. triage .....	211
7.17. triage store .....	211
7.18. version .....	211

Definitions of terms used in this API documentation.

### 7.1. Attribute

#### 7.1.1.

Property of given CID used during the triage process to specify the state of the CID (for example, its classification, severity, or the type of action that it requires). Depending on its attribute type, an attribute can take a string value or a value from a list of values. To retrieve a complete list, you can invoke `getAttributes()`.

### 7.2. Checker

#### 7.2.1.

A program that traverses the paths in your source code to find specific coding issues in it. Examples include `RACE_CONDITION`, `RESOURCE_LEAK`, and `INFINITE_LOOP`. Issues can pertain to software quality and security, and to policy violations in developer tests.

### 7.3. CID

#### 7.3.1.

Unique identifier (Coverity ID) for one or more instances (occurrences) of a given software issue. The API often uses *MergedDefect* to refer to a CID.

Issue instances, both within a snapshot and across snapshots (even in different streams), are grouped together according to similarity, with the intent that two issues are "similar" if the same source code change would fix them both. Such instances share the same CID. Coverity Connect associates triage data, such as classification, action, and severity, with the CID (rather than with an individual issue).

## **7.4. component**

### **7.4.1.**

A named grouping of source code files to which role-based access control (RBAC) permissions can be applied. Each component belongs to a component map.

## **7.5. defect**

### **7.5.1.**

A software issue found through a Coverity analysis. (See CID).

## **7.6. external (ext.) reference**

### **7.6.1.**

Attribute typically used to enter a code or ID used by your company internally to track an issue, for example, through its Bug tracking system. This value helps you map a CID to the internal code.

## **7.7. fix target**

### **7.7.1.**

Attribute typically used to designate a version of the source code in which a CID is scheduled to be fixed.

## **7.8. permission**

### **7.8.1.**

Access rule intended primarily for administrative tasks, such as server configuration, user and group creation and management, and so forth. A permission is assigned indirectly to a user or group through a role with which the permission is associated. To retrieve a list of permissions, see `getAllPermissions()`. See also `Permission Values`.

## **7.9. project**

### **7.9.1.**

Container for one or more streams.

## 7.10. role

### 7.10.1.

A named set of one or more permissions that a user or group can have. Some roles are built-in and uneditable (for example, System Admin and Visitor). Others built-in roles are "custom" (or customizable) in that they are editable (for example, Developer, Desktop Developer, and Committer). It is also possible to create custom roles.

Roles can be assigned to users and groups at the global level, project level, stream level, triage store level, and component level. For more detail on roles, see the section on RBAC (role-based access control) in the Coverity Platform Use and Administration Guide.

## 7.11. snapshot

### 7.11.1.

A copy of the state of a code base at the date and time it was analyzed. A snapshot includes information about software issues found by the analysis and the source code in which the issues were found.

## 7.12. source version

### 7.12.1.

Optional version property of a snapshot that typically identifies the version of the source code that was analyzed. This property is present only if passed when committing analysis results (passed by the `--version` option of `cov-commit-defects` command or through an advanced commit setting in Coverity Wizard).

## 7.13. stream

### 7.13.1.

In Coverity Connect, a sequential set of snapshots produced by successive analyses of a given code base.

## 7.14. subcategory

### 7.14.1.

Subcategory of a checker. Used to help identify the issues that the checker finds. Examples include `pointer_deref_write` for an UNINIT checker and `deref_constant_null` for a FORWARD\_NULL checker. Subcategory value can be `none`.

## 7.15. target

### 7.15.1.

Optional target property of a snapshot that typically identifies the target platform for the source code that was analyzed. This property is present only if passed when committing analysis results (passed by the `--target` option of the `cov-commit-defects` command or through an advanced commit setting in Coverity Wizard).

## 7.16. triage

### 7.16.1.

The process of identifying and tracking important information about a CID. Built-in, but configurable, triage attributes include Classification, Severity, Action, Ext. Reference, and Owner. To triage an issue, a developer might change the value of Classification attribute from Unclassified to Bug. Developers can also triage a CID by adding comments about it.

## 7.17. triage store

### 7.17.1.

Repository for the current and historical triage values of CIDs. The state of triage values for a CID changes, for example, when a developer performs triage.

## 7.18. version

### 7.18.1.

\* Internal or external version of Coverity Platform. See `getVersion()`.

\* Internal or external version of an analysis or build associated with a snapshot. See `snapshotInfoDataObject`.

\* Version of the source code in a snapshot. See `source version`.

---

## Chapter 8. API History

### Table of Contents

8.1. Change Logs .....	212
8.2. Configuration Service Log .....	213
8.3. Defect Service Log .....	221

### 8.1. Change Logs

Change logs since version 3 (v3) for the following services are available:

- Section 8.2, “Configuration Service Log”
- Section 8.3, “Defect Service Log”

Section 8.3, “Defect Service Log”

The APIs follow a versioning scheme that differs from the Coverity installation versions. The following table lists the Coverity Connect (previously called Integrity Manager) version in which each version of the API was released. In general, you should use the most current version of the API.

Typically, the API Version will reach End of Life in the next major release after it is Deprecated. For example, v3 was deprecated in Coverity Connect version 6.5.1, and reached End of Life in Coverity Connect version 6.6.

API Version	Released in Coverity Connect	Deprecated and EOL in Coverity Connect
v1	version 5.2	Deprecated: version 6.0 End of life: version 6.5
v2	version 5.3	Deprecated: version 6.0 End of life: version 6.5
v3	version 5.4.1	Deprecated: version 6.5.1 End of life: version 6.6
v4	version 5.5.1	Deprecated: version 7.5 End of life: version 7.6
v5	version 6.0.3	Deprecated: version 7.6 End of life: version 7.7
v6	version 6.5.0	Deprecated: version 7.7 End of life: version 2019.12
v7	version 6.5.1, modified in 6.6	Deprecated: version 2019.12 End of life: version 2020.03

---

API Version	Released in Coverity Connect	Deprecated and EOL in Coverity Connect
-------------	------------------------------	--

---

v8	version 7.0, modified in 7.5, 7.5.1	Deprecated: version 2019.12 End of life: version 2020.03
----	-------------------------------------	---

v9	version 7.6.0	
----	---------------	--

---

## 8.2. Change Log: Configuration Service

### 8.2.1.

Change log for the following **Configuration Service** items:

- Section 8.2.2, “Operations: Configuration Service”
- Section 8.2.3, “Data Objects” (Complex Types)

### 8.2.2. Operations: Configuration Service

#### 8.2.2.1.

Change log for operations in the **Configuration Service**. Changes to *fields* are logged from v6 through the latest version only. Field modifications in earlier versions are not logged at this time. Note that v7 of the API has changed between Coverity Connect versions 6.5.1 and 6.6. The v7 values in the table make this distinction by using v7 (6.6) and v7 (6.5.1).

Configuration Operations	Introduced	Modified	Removed	Note
getActions	< or = v3		v4	REMOVED.
copyStream	v4	<b>v8 (7.5),</b> v8 (7.0)		v8 (7.5): New return fields: enableDesktopAnalysis, summaryExpirationDays, analysisVersionOverride, pluginVersionOverride, versionMismatchMessage  v8 (7.0): New return fiel outdated
createAttribute	v4			
createComponentMap	< or = v3			
createGroup	< or = v3	v4		Created in Admin Service.  v4: Moved to Configuration service.
createLdapConfiguration	v7 (6.6)			
createProject	v4			
createRole	v4			
createStream	v4			

---

## API History

---

Configuration Operations	Introduced	Modified	Removed	Note
createStreamInProject	v4	v8 (7.0)		v8 (7.0) (new return field): outdated
createTriageStore	v5			
createUser	< or = v3	v4		Created in Admin Service. v4: Moved to Configuration service.
deleteAttribute	v4			
deleteComponentMap	v4			
deleteGroup	< or = v3	v4		Created in Admin Service. v4: Moved to Configuration service.
deleteLdapConfiguration	v7 (6.6)			
deleteProject	v4			
deleteRole	v4			
deleteSnapshot	v4			
deleteStream	v4	v6		v6 (new fields): onlyIfEmpty
deleteTriageStore	v5			
deleteUser	< or = v3	v4		Created in Admin Service. v4: Moved to Configuration service.
executeNotification	v7 (6.6)			
getAllDomains	v4		v5	REMOVED.
getAllGroups	< or = v3	v4	v4	REMOVED/RENAMED. Created in Admin Service. v4: Renamed getGroups
getAllIntegrityControlPermissions	v5		v7 (6.6)	REMOVED.
getAllLdapConfigurations	v7 (6.6)			
getAllPermissions	v4			
getAllRoles	< or = v3	v4		Created in Admin Service. v4: Moved to Configuration service.
getArchitectureAnalysisConfiguration	v8 (7.5)			New in v8 (7.5)
getAssignableUsers	< or = v3		v4	REMOVED.
getAttribute	v4			
getAttributes	v4			
getBackupConfiguration	v8 (7.0)			New in v8 (7.0).
getCategoryNames	v9			

## API History

---

Configuration Operations	Introduced	Modified	Removed	Note
getCheckerNames	v9			
getCheckerProperties	v4		v9	REMOVED.  The Defect Service, <code>getStreamDefects</code> , replaces <code>getCheckerProperties.getStreamDefects</code> returns the <code>DefectInstanceDataObj</code> , which contains all necessary checker properties. The other potential use case for <code>getCheckerProperties</code> was to receive a list of all categories, types, and checker names available for filtering. This can be accomplished with the following operations: <code>getCategoryNames</code> , <code>getCheckerNames</code> , <code>getTypeNames</code>
getClassifications	< or = v3		v4	REMOVED.
getCommitState	v5			
getComponent	v4			
getComponentMaps	v4			
getDefectStatuses	v4			
getDeleteSnapshotJobInfo	v7 (6.5.1)			v7 (new fields): <code>snapshotId</code>  v7 (new response): <code>getDeleteSnapshotJobInfoResponse</code>
getDeveloperStreamsProjects	v8			
getGroup	< or = v3	v4		Created in Admin Service.  v4: Moved to Configuration service.
getGroups	v4			Replaced <code>getAllGroups</code> .
getLastUpdateTimes	v6		v9	REMOVED.
getLdapServerDomains	v6			
getLicenseConfiguration	v8 (7.0)			New in v8 (7.0).
getLicenseState	v5			
getLoggingConfiguration	v8 (7.0)			New in v8 (7.0).
getMessageOfTheDay	v5			
getProjects	v4			
getRole	v4			
getServerTime	< or = v3	v4		Created in Admin Service.

## API History

Configuration Operations	Introduced	Modified	Removed	Note
				v4: Moved to Configuration service.
getSeverities	< or = v3		v4	REMOVED.
getSignInConfiguration	v8 (7.0)			New in v8 (7.0).
getSkeletonizationConfiguration	v8 (7.0)			New in v8 (7.0).
getSnapshotInformation	v4	<b>v8 (7.5)</b>		v8 (7.5): New return fields: portableAnalysisSettings, codeVersionDate, hasSummaries
getSnapshotPurgeDetails	v7 (6.5.1), v7(6.6)	v8 (7.0)		Deprecated in v8 (7.0): Use getSkeletonizationConfiguration() instead.  First documented in v7 (6.6) but first available in the WSDL file for the 6.5.1 release.
getSnapshotsForStream	v4			
getStandardAttribute	v9			
getStandardAttributes	v9			
getStreams	v4	v8 (7.0), <b>v8 (7.5)</b>		v8 (7.5): New return fields: enableDesktopAnalysis, summaryExpirationDays, analysisVersionOverride, pluginVersionOverride, versionMismatchMessage  v8 (7.0): New return field outdated
getSystemConfig	v4			
getTriageStores	v5			
getTypeNames	v9			
getUser	< or = v3	v4		Created in Admin Service.
getUsers	v4			v4: Moved to Configuration service. Replaced getUsersForGroup.
getUsersForGroup	< or = v3	v4	v4	REMOVED/REPLACED.  Created in Admin Service.  v4: Replaced by getUsers in v4.
getVersion	v6			
importLicense	v8 (7.0)			New in v8 (7.0). Passes v8 licenseSpecDataObj.
notify	< or = v3	v4		Created in Admin Service.

## API History

Configuration Operations	Introduced	Modified	Removed	Note
purgeSnapshotDetails	v7 (6.5.1)	v7 (6.6)		v4: Moved to Configuration service. v7 (6.6): snapshotId field removed. First documented in v7 (6.6) but first available in the WSDL file for the 6.5.1 release.  <b>Deprecated</b> in v8 (7.0). Use <code>getSkeletonizationConfiguration()</code> instead.
setAcceptingNewCommits	v5			
setArchitectureAnalysisConfiguration	v8 (7.5)			New in v8 (7.5)
setBackupConfiguration	v8 (7.0)			New in v8 (7.0). Passes v8 <code>backupConfigurationDataObj</code> .
setLoggingConfiguration	v8 (7.0)			New in v8 (7.0). Passes v8 <code>loggingConfigurationDataObj</code> .
setMessageOfTheDay	v5	v7 (6.5.1)		v7 (new field name): <code>message</code> replaces <code>arg0</code>
setSkeletonizationConfiguration	v8 (7.0)			New in v8 (7.0). Replaces <code>setSnapshotPurgeDetails()</code> . Passes v8 <code>signInSettingsDataObj</code> .
setSnapshotPurgeDetails	v7 (6.6)	v8 (7.0)		Deprecated in v8 (7.0). Use <code>setSkeletonizationConfiguration()</code> instead.
updateActions	< or = v3		v4	REMOVED.
updateAttribute	v4			
updateComponentMap	v4			
updateGroup	v4			
updateLdapConfiguration	v7 (6.6)			
updateProject	v4			
updateRole	v4			
updateSeverities	< or = v3		v4	REMOVED.
updateSignInConfiguration		v8 (8.7.0)		v8 (8.7.0): Added <code>signInSettingsDataObj.authenticationMethod</code>
		v9		Removed <code>enableSessionTimeout</code> parameter.
updateSnapshotInfo	v6			
updateStream	v4	v8 (7.0)		v8 (7.0) (new return field): outdated
updateTriageStore	v5			
updateUser	v5			

## 8.2.3. Data Objects: Configuration Service

### 8.2.3.1.

Change log for data objects in the **Configuration Service**. Changes to *fields* are logged from v6 through the latest version only. Field modifications in earlier versions are not logged at this time. Note that v7 of the API has changed between Coverity Connect versions 6.5.1 and 6.6. The v7 values in the table make this distinction by using v7 (6.6) and v7 (6.5.1).

Data Object	Introduced	Modified	Removed	Notes
attributeDefinitionDataObj	v4			
attributeDefinitionIdDataObj	v4			
attributeDefinitionSpecDataObj	v4			
attributeValueChangeSpecDataObj	v4			
attributeValueDataObj	v4	v7 (6.5.1)		v7 (new fields): issueKindList
attributeValueIdDataObj	v4			
attributeValueSpecDataObj	v4			
checkerPropertyDataObj	v4	v7 (6.5.1)		v7 (new fields): eventSetCaptionsv7 (removed fields): eventSet0Caption, eventSet1Caption, eventSet2Caption
checkerPropertyFilterSpecDataObj	v4		v9	REMOVED.
checkerSubcategoryIdDataObj	v4			
commitStateDataObj	v5			
componentDataObj	< or = v3		v7 (6.5.1)	v7 (new fields): roleAssignments (a roleAssignmentDataObj)  v7 (removed fields): groupPermissions (a groupPermissionsDataObj)
componentDefectRuleDataObj	< or = v3			
componentIdDataObj	< or = v3			
componentMapDataObj	< or = v3			
componentMapFilterSpecDataObj	< or = v3			
componentMapIdDataObj	< or = v3			
componentMapSpecDataObj	< or = v3			
componentPathRuleDataObj	< or = v3			

## API History

Data Object	Introduced	Modified	Removed	Notes
configurationDataObj	v4	v7, v9		v7 (removed fields): groupPermissions (a groupPermissionsDataObj)
deleteSnapshotJobInfoDataObj				v9 (new field): issueExportUrl
featureUpdateTimeDataObj	v6			v7 (fields): snapshotId, status
groupDataObj	< or = v3	v6		Appeared v3 Configuration and Admin Service.  v6: domainName field renamed <i>domain</i>
groupFilterSpecDataObj	< or = v3			
groupIdDataObj	< or = v3	v7 (6.6)		Appeared in v3 Configuration and Admin Service.  New field in v7 (6.6): displayName
groupPermissionDataObj	< or = v3		v7 (6.5.1)	REMOVED.  Contained the groupId and groupRole fields.
groupsPageDataObj	< or = v3			Introduced in Admin Service.
groupSpecDataObj	< or = v3	v6		Introduced in Admin Service.  v6: domainName field renamed <i>domain</i>
ldapConfigurationDataObj	v7 (6.6)			
ldapConfigurationSpecDataObj	v7 (6.6)			
licenseStateDataObj	v5			
localizedValueDataObj	v9			displayName, name
pageSpecDataObj	< or = v3			Introduced in Admin Service.
permissionDataObj	v4			
projectDataObj	< or = v3	v6		v6 (removed fields): defaultTriageScope
projectFilterSpecDataObj	< or = v3	v7 (6.6)		New field in v7 (6.6): includeChildren, includeStreams
projectIdDataObj	< or = v3			

## API History

<b>Data Object</b>	<b>Introduced</b>	<b>Modified</b>	<b>Removed</b>	<b>Notes</b>
projectFilterSpecDataObj	< or = v3	v6		v6 (removed fields): defaultTriageScope
projectIdDataObj	< or = v3			
projectSpecDataObj	< or = v3			
roleAssignmentDataObj	v4			
roleDataObj	< or = v3			Introduced in Admin Service.
roleIdDataObj	v4			
roleSpecDataObj	v4			
serverDomainIdDataObj	v4			
signInSettingsDataObj		v8 (8.7.0)		v8 (8.7.0): Added authenticationMethod field.
		v9		Removed enableSessionTimeout parameter.
snapshotFilterSpecDataObj	< or = v3	v8 (7.5)		v8 (7.5). New fields: hasSummaries, lastBeforeCodeVersionDate
snapshotIdDataObj	< or = v3			
snapshotInfoDataObj	< or = v3	v6, v8 (7.0)		v8 (7.0) typo fix: Misspelled field name aysisIntermediateDir is now analysisIntermediateDir
snapshotPurgeDetailsObj	v7 (6.6)			
standardAttributeDataObj	v9			
standardAttributeIdDataObj	v9			
standardAttributeValueDataObj	v9			
standardAttributeValueIdDataObj	v9			
streamDataObj	< or = v3			v6 (new fields): autoDeleteOnExpiry
streamFilterSpecDataObj	< or = v3			
streamIdDataObj	< or = v3			
streamSpecDataObj	< or = v3	v6, v7 (6.6), v8 (7.5)		v8 (7.5): New fields: enableDesktopAnalysis, summaryExpirationDays, analysisVersionOverride, pluginVersionOverride, versionMismatchMessage

Data Object	Introduced	Modified	Removed	Notes
				v6 (new fields): autoDeleteOnExpiry
triageStoreDataObj	v5	v7 (6.5.1)		New field in v7 (6.6): allowCommitWithoutPassword
triageStoreFilterSpecDataObj	v5	v7 (6.6)		v7 (new fields): streamIds New field in v7 (6.6): roleAssignments
triageStoreIdDataObj	v5	v7 (6.6)		New field in v7 (6.6): roleAssignments
triageStoreSpecDataObj	v5			
userDataObj	< or = v3	v6		Introduced in Admin Service.
userFilterSpecDataObj	v4	v7 (6.6)		v6 (new fields): superUser New field in v7 (6.6): includeDetails
usersPageDataObj	< or = v3			Introduced in Admin Service.
userSpecDataObj	< or = v3			Introduced in Admin Service.
versionDataObj	v6			

## 8.3. Change Log: Defect Service

### 8.3.1.

Change log for the following **Defect Service** items:

- Section 8.3.2, “Operations”
- Section 8.3.3, “Data Objects” (Complex Types)

### 8.3.2. Operations: Defect Service

#### 8.3.2.1.

Change log for operations in the **Defect Service**. Changes to *fields* are logged as noted at this time. Also note that v7 of the API has changed between Coverity Connect versions 6.5.1 and 6.6. The v7 values in the table make this distinction by using v7 (6.6) and v7 (6.5.1).

Operations	Introduced	Modified	Removed	Note
copyStreamDefectStates	v4		v5	REMOVED.
getCheckerSubcategoriesForProject	v4		v9	REMOVED.
getCheckerSubcategoriesForStreams	v4		v9	REMOVED.
getCIDsForProject	< or = v3		v8 (7.0)	REMOVED.

## API History

Operations	Introduced	Modified	Removed	Note
getCIDsForStreams	< or = v3		v8 (7.0)	REMOVED.
getComponentMetricsForProject	< or = v3			
getFileContents	< or = v3			
getMergedDefectHistory	< or = v3	Yes (some version info not available at this time),  v8 (7.0)		<i>Changed fields since introduction:</i> scopePattern removed.  streamIdDataObj.name added.  <i>Changed response since introduction:</i> mergedDefectDataObj  v8 (7.0) changes: Passes v8 snapshotScopeSpecDataObj. Now returns getMergedDefectsForStreams, streamExcludeQualifier, streamIncludeNameList, and streamIncludeQualifier.
getMergedDefectDetection	v8 (7.0)			New in v8 (7.0).Passes v8 mergedDefectIdDataObj.
getMergedDefectsForProject	< or = v3		v8 (7.0)	REMOVED. Replaced by  getMergedDefectsForProjectScope.
getMergedDefectsForProjectScope	v8 (7.0)			New in v8 (7.0).Replaces getMergedDefectsForProject.
getMergedDefectsForSnapshotScope	v8 (7.0)			New in v8 (7.0).Passes v8 snapshotScopeDefectFilterSpecDataObj and snapshotScopeSpecDataObj.
getMergedDefectsForStreams	< or = v3			
getStreamDefects	< or = v3	Yes (version information not available at this time)		<i>Changed fields since introduction:</i> streamDefectFilterSpecDataObj.streamIdList added  scopePattern removed
getTrendRecordsForProject	< or = v3			

## API History

Operations	Introduced	Modified	Removed	Note
updateDefectInstanceProperties	v3			<b>v8 (7.5):</b> Do not use this operation.
updateStreamDefects	< or = v3			
updateTriageForCIDsInTriageStore	v8 (7.0)	v8 (7.0)		v8 (7.0) changes: Passes v8 mergedDefectIdDataObj
getTriageHistory	v8 (7.0)			New in v8 (7.0). Passes v8 mergedDefectIdDataObj.

### 8.3.3. Data Objects: Defect Service

#### 8.3.3.1.

Change log for data objects in the **Defect Service**. Changes to *fields* are logged from v6 through the latest version only. Field modifications in earlier versions are not logged at this time. Note that v7 of the API has changed between Coverity Connect versions 6.5.1 and 6.6. The v7 values in the table make this distinction by using v7 (6.6) and v7 (6.5.1).

Data Objects	Introduced	Modified	Removed	Note
attributeDefinitionDataObj	v4		v5	REMOVED.
attributeDefinitionIdDataObj	v4			
attributeDefinitionValueFilterMapDataObj	v4			
attributeValueDataObj	v4		v5	REMOVED.
attributeValueIdDataObj	v4			
checkerPropertyDataObj	v4		v5	REMOVED.
checkerFilterSpecDataObj	< or = v3		v4	REMOVED.
checkerSubcategoryFilterSpecDataObj	v8 (7.0)		v9	REMOVED.
checkerSubcategoryIdDataObj	v8 (7.0)		v9	REMOVED.
componentIdDataObj	< or = v3			
componentMetricsDataObj	< or = v3	v8 (7.0)		v8 (7.0) (removed fields): inspectedCount, resolvedCount
defectChangeDataObj	< or = v3	v6, v7 (6.5.1)		v6 (new fields): fixTargetChange  v7 (new fields): attributeChanges  v7 (removed fields): actionChange, customAttributeChanges,

## API History

Data Objects	Introduced	Modified	Removed	Note
				externalReferenceChange, fixTargetChange, ownerChange, severityChange, statusChange
defectDetectionHistoryDataObj	v7.0)			
defectInstanceDataObj	< or = v3	v9		MODIFIED since introduction:  Added checkerSubcategoryIdDataObj.  Removed checkerSubcategory.  v9 (new fields): category, checkerName, component, cwe, domain, eventSetCaptions, impact, issueKinds, localEffect, longDescription, type
defectInstanceIdDataObj	< or = v3			v8 (7.5): Do not use this data object.
defectStateCustomAttributeValueDataObj	v4		v7 (6.5.1)	REMOVED.
defectStateDataObj	< or = v3			Modified since introduction.
defectStateAttributeValueDataObj	v7 (6.5.1)			v7 (new fields): attributeDefinitionId, attributeValueId
defectStateCustomAttributeValueDataObj	v4		v7 (6.5.1)	v7 (removed fields): attributeDefinitionId, attributeValueId
defectStateDataObj	v4	v6, v7 (6.5.1)		v6 (new fields): fixTarget  v7 (new fields): defectStateAttributeValues
				v7 (removed fields): action, classification, comment, externalReference, owner, severity, status, defectStateCustomAttributeValues, fixTarget
defectStateSpecDataObj	v4	v6, v7 (6.5.1)		v6 (new fields): fixTarget

## API History

---

Data Objects	Introduced	Modified	Removed	Note
				v7 (new fields): defectStateAttributeValues
				v7 (removed fields): action, classification, comment, externalReference, owner, severity, status, defectStateCustomAttributeValues, fixTarget
eventDataObj	< or = v3			Modified since introduction:  Added <i>fileId</i> (a fileIdDataObj) and <i>moreInformationId</i> .  Removed <i>file</i> .
fieldChangeDataObj	< or = v3			
fileContentsDataObj	v4			
fieldDataObj	v4			
functionInfoDataObj	< or = v3			Modified since introduction:  Removed filePathname.  Added <i>fileId</i> (a fileIdDataObj).
localizedValueDataObj	v9			
mergedDefectDataObj	< or = v3	v6, v7 (6.5.1), v8 (7.5.1),  v9		v6 (new field): ownerName  v7 (new field): defectStateAttributeValues  v7 (removed fields): action, classification,comment, defectStateCustomAttributeValues, externalReference, fixTarget, owner, ownerName, translatedOwner, severity, status  v8 (new field in 7.5.1): firstDetectedBy  v9 (new fields):cwe, displayCategory,

## API History

Data Objects	Introduced	Modified	Removed	Note
				displayImpact, displayIssueKind, displayType,issueKind
mergedDefectFilterSpecDataObj	v3	v6, v7(6.5.1), v7 (6.6),v8 (7.0), v9		v9 (removed field): checkerSubcategory v6 (new fields): fixTargetNameList v7 (6.5.1) new fields: issueKindList, ownerNamePattern v7 (6.6): legacyNameList v8 (7.0) (removed fields):streamSnapshotFilterSpecIncludeList, streamSnapshotFilterSpecExcludeList, streamSnapshotExcludeAll, streamSnapshotIncludeAll v8 (7.5): Do not use the following fields to this data object: defectPropertyKey, defectPropertyPattern v9 (removed field): checkerSubcategoryFilterSpecList v9 (new fields): checkerCategoryList, checkerList, checkerTypeList, cweList, impactList
mergedDefectIdDataObj	v8 (7.0)			v8 (7.0). Replaces cid parameter to getMergedDefectHistory and cids parameter to getStreamDefects.
mergedDefectsPageDataObj	v3			
pageSpecDataObj	< or = v3			
projectIdDataObj	< or = v3			
projectMetricsDataObj	< or = v3			
projectScopeDefectFilterSpecDataObj	v3	v8 (7.5.1)		v8 (new field in 7.5.1): firstDetectedBy

## API History

Data Objects	Introduced	Modified	Removed	Note
projectTrendRecordFilterSpecDataObj				
propertyDataObj	< or = v3			
propertySpecDataObj	< or = v3			v8 (7.5):Do not use this data object.
snapshotIdDataObj	< or = v3		v8 (7.0)	REMOVED from Defect service only.
snapshotScopeDefectFilterSpecDataObj				
snapshotScopeSpecDataObj				
standardAttributeIdDataObj	v9			
standardAttributeValueFilterMapDataObj	v9			
standardAttributeValueIdDataObj				
streamDefectDataObj	< or = v3	v6, v7 (6.5.1), v9		v6 (new fields): fixTarget  v7 (new fields): defectStateAttributeValues  v7 (removed fields): action, classification, comment, externalReference, owner, severity, status, defectStateCustomAttributeValues, fixTarget  v9 (removed field): checkerSubcategoryId  v9 (new fields): checkerName, domain
streamDefectFilterSpecDataObj	< or = v3			
streamDefectIdDataObj	< or = v3			
streamFunctionDataObj	v6			
streamFunctionPageDataObj	v6			
streamIdDataObj	< or = v3			
streamSnapshotFilterSpecDataObj			v8 (7.0)	REMOVED.
trriageStoreIdDataObj	v6			

---

# Appendix A. Legal Notice

## Table of Contents

A.1. Legal Notice .....	228
-------------------------	-----

### A.1. Legal Notice

Copyright © 2020 Synopsys, Inc. All rights reserved worldwide.

The information contained in this document, and the Licensed Product provided by Synopsys, are the proprietary and confidential information of Synopsys, Inc. and its affiliates and licensors, and are supplied subject to, and may be used only by Synopsys customers in accordance with the terms and conditions of a license agreement previously accepted by Synopsys and that customer. Synopsys' current standard end user license terms and conditions are contained in the cov\_EULM files located at <install\_dir>/doc/en/licenses/end\_user\_license. Portions of the product described in this documentation use third-party material. Notices, terms and conditions, and copyrights regarding third party material may be found in the <install\_dir>/doc/en/licenses directory. Customer acknowledges that the use of Synopsys Licensed Products may be enabled by authorization keys supplied by Synopsys for a limited licensed period. At the end of this period, the authorization key will expire. You agree not to take any action to work around or override these license restrictions or use the Licensed Products beyond the licensed period. Any attempt to do so will be considered an infringement of intellectual property rights that may be subject to legal action. If Synopsys has authorized you, either in this documentation or pursuant to a separate mutually accepted license agreement, to distribute Java source that contains Synopsys annotations, then your distribution should include Synopsys' analysis\_install\_dir/library/annotations.jar to ensure a clean compilation. This annotations.jar file contains proprietary intellectual property owned by Synopsys. Synopsys customers with a valid license to Synopsys' Licensed Products are permitted to distribute this JAR file with source that has been analyzed by Synopsys' Licensed Products consistent with the terms of such valid license issued by Synopsys. Any authorized distribution must include the following copyright notice: Copyright © 2017 Synopsys, Inc. All rights reserved worldwide. U.S. GOVERNMENT RESTRICTED RIGHTS: The Software and associated documentation are provided with Restricted Rights. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in subparagraph (c)(1) of The Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of Commercial Computer Software – Restricted Rights at 48 CFR 52.227-19, as applicable. The Manufacturer is: Synopsys, Inc. 690 E. Middlefield Road, Mountain View, California 94043. The Licensed Product known as Coverity is protected by multiple patents and patents pending, including U.S. Patent No. 7,340,726. Trademark Statement Coverity and the Coverity logo are trademarks or registered trademarks of Synopsys, Inc. in the U.S. and other countries. Synopsys' trademarks may be used publicly only with permission from Synopsys. Fair use of Synopsys' trademarks in advertising and promotion of Synopsys' Licensed Products requires proper acknowledgement. Microsoft, Visual Studio, and Visual C# are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Microsoft Research Detours Package, Version 3.0. Copyright © Microsoft Corporation. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or affiliates. Other names may be trademarks of their respective owners. "MISRA", "MISRA C" and the MISRA triangle logo are registered trademarks of MISRA Ltd, held on behalf of the MISRA Consortium. © MIRA Ltd, 1998 - 2013. All rights reserved. The name FindBugs™ and the FindBugs logo are trademarked by The University of Maryland. Other names and brands may

be claimed as the property of others. This Licensed Product contains open source or community source software ("Open Source Software") provided under separate license terms (the "Open Source License Terms"), as described in the applicable license agreement under which this Licensed Product is licensed ("Agreement"). The applicable Open Source License Terms are identified in a directory named "Licenses" provided with the delivery of this Licensed Product. For all Open Source Software subject to the terms of an LGPL license, Customer may contact Synopsys at [software-integrity-support@synopsys.com](mailto:software-integrity-support@synopsys.com) and Synopsys will comply with the terms of the LGPL by delivering to Customer the applicable requested Open Source Software package, and any modifications to such Open Source Software package, in source format, under the applicable LGPL license. Any Open Source Software subject to the terms and conditions of the GPLv3 license as its Open Source License Terms that is provided with this Licensed Product is provided as a mere aggregation of GPL code with Synopsys' proprietary code, pursuant to Section 5 of GPLv3. Such Open Source Software is a self-contained program separate and apart from the Synopsys code that does not interact with the Synopsys proprietary code. Accordingly, the GPL code and the Synopsys proprietary code that make up this Licensed Product co-exist on the same media, but do not operate together. Customer may contact Synopsys at [software-integrity-support@synopsys.com](mailto:software-integrity-support@synopsys.com) and Synopsys will comply with the terms of the GPL by delivering to Customer the applicable requested Open Source Software package in source code format, in accordance with the terms and conditions of the GPLv3 license. No Synopsys proprietary code that Synopsys chooses to provide to Customer will be provided in source code form; it will be provided in executable form only. Any Customer changes to the Licensed Product (including the Open Source Software) will void all Synopsys obligations under the Agreement, including but not limited to warranty, maintenance services and infringement indemnity obligations. The Cobertura package, licensed under the GPLv2, has been modified as of release 7.0.3. The package is a self-contained program, separate and apart from Synopsys code that does not interact with the Synopsys proprietary code. The Cobertura package and the Synopsys proprietary code co-exist on the same media, but do not operate together. Customer may contact Synopsys at [software-integrity-support@synopsys.com](mailto:software-integrity-support@synopsys.com) and Synopsys will comply with the terms of the GPL by delivering to Customer the applicable requested open source package in source format, under the GPLv2 license. Any Synopsys proprietary code that Synopsys chooses to provide to Customer upon its request will be provided in object form only. Any changes to the Licensed Product will void all Coverity obligations under the Agreement, including but not limited to warranty, maintenance services and infringement indemnity obligations. If Customer does not have the modified Cobertura package, Synopsys recommends to use of the JaCoCo package instead. For information about using JaCoCo, see the description for `cov-build --java-coverage` in the Command and Ant Task Reference.

LLVM/Clang subproject Copyright © All rights reserved. Developed by: University of Illinois at Urbana-Champaign, Computer Science Department . Permission is hereby granted, free of charge, to any person obtaining a copy of LLVM/Clang and associated documentation files ("Clang"), to deal with Clang without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of Clang, and to permit persons to whom Clang is furnished to do so, subject to the following conditions: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution. Neither the name of the University of Illinois at Urbana-Champaign, nor the names of its contributors may be used to endorse or promote products derived from Clang without specific prior written permission. CLANG IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING

## Legal Notice

---

FROM, OUT OF OR IN CONNECTION WITH CLANG OR THE USE OR OTHER DEALINGS WITH CLANG. Rackspace Threading Library (2.0) Copyright © Rackspace, US Inc. All rights reserved. Licensed under the Apache License, Version 2.0 (the "License"); you may not use these files 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. SIL Open Font Library subproject Copyright © 2017, Synopsys Inc. All rights reserved worldwide. ([www.synopsys.com](http://www.synopsys.com)), with Reserved Font Name fa-gear, fa-info-circle, fa-question. This Font Software is licensed under the SIL Open Font License, Version 1.1. This license is available with a FAQ at <http://scripts.sil.org/OFL>.