SOLA Resource Manager User's Guide Version 6.4.2

WS-Security
Standards-based security enforcement & management
Support for architectural standards
Global dictionary, WSDL first
Management console & dashboard
SLA, monitoring, logging & audit controls
WS-Policy
Standards-based policy enforcement & management
High performance SOAP and REST API engine
Web 2.0 development environment
Change & release management
Version control, backups & migration
Orchestration & microflows
Registry
UDDI, WSRR

Revision Date: August 2017

Contents

About SOLA Resource Manager

Resource Manager is a run-time governance tool for managing SOLA and controlling SOLA's run-time authentication and authorization to resources (programs, methods, etc.) running in specific SOLA Containers (regions). Resource Manager also offers additional capabilities, such policy governance and dashboard monitoring of entire containers, single programs or methods.

As Resource Manager is only concerned with run-time access, all access rights granted by Resource Manager are "invoke" only. This means that if a user or IP address is given access to a program, that access covers that user's or IP addresses right to run the program and nothing else. Development governance is handled by SOLA Developer.

Resource Manager is a web 2.0 application that features a rich graphical interface, drag and drop capabilities and a tab based workspace. Being a web application, Resource Manager does not require an installation on the user's workstation and can be accessed by any machine with a web browser and sufficient access rights.

A note on terminology: prior versions of SOLA only offered a CICS deployment option, but starting with version 6.0, SOLA had the ability to integrate with IMS from a CICS container. With the release of the SOLA IMS plug-in, we have removed the CICS dependency and are now shipping a Started Task Address Space version of the SOLA runtime (managed through Resource Manager). For this reason, we will no longer refer to the SOLA runtime a "SOLA TOR." It will henceforth be referred to as a "SOLA Container."

Resource Manager Basics

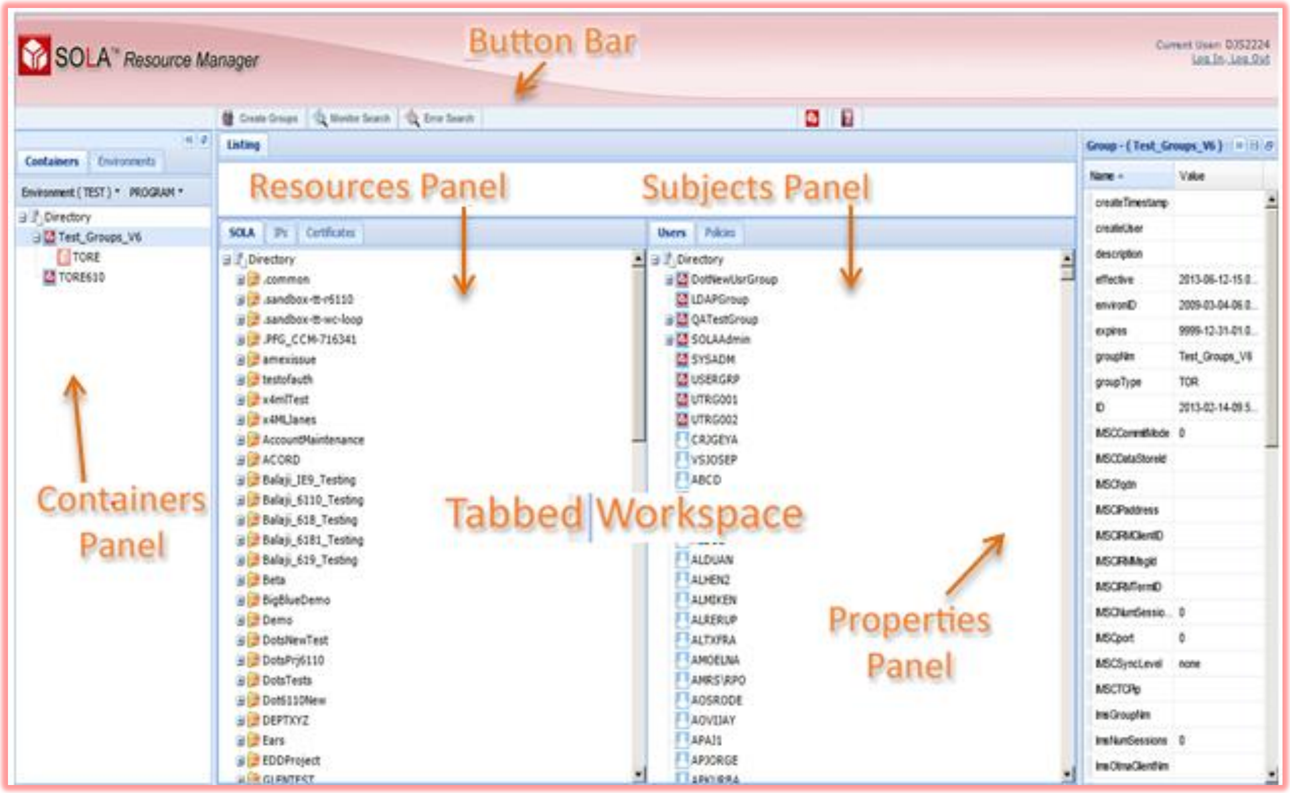
SOLA Resource Manager is a browser based run-time administration application that can be used to control various run time access controls such as user access, IP security, policies and more.

The Resource Manager Window

The Resource Manager window is divided into several parts, illustrated in the figure below.

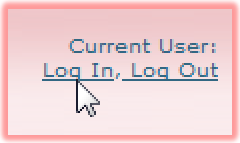
Some of the panels in Resource Manager can be minimized by using the minimize buttons («). When working on low resolution displays, minimizing panels can maximize your workspace.

Resource Manager provides a drag and drop capability to allow you to quickly and easily move items from one group to another, or from one tree to another



Name ▾	Value
createTimestamp	
createUser	
description	
effective	2013-06-12-15.0...
environID	2009-03-04-06.0...
expires	9999-12-31-01.0...
groupNm	Test_Groups_V6
groupType	TOR
ID	2013-02-14-09.5...
MSCCommitMode	0
MSCDataStoreId	
MSCfqdn	
MSCIPaddress	
MSCIRMClientID	
MSCIRMMsgId	
MSCIRMTermID	
MSCNumSessio...	0
MSCport	0
MSCSyncLevel	none
MSCTCPIp	
ImsGroupNm	
ImsNumSessions	0
ImsOtmaClientNm	

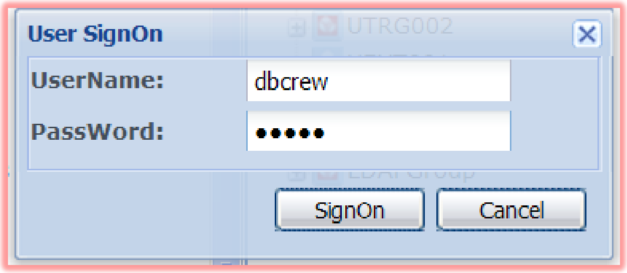
Logging In and User Properties



Current User:
Log In, Log Out

Most of the functionality of Resource Manager is restricted to authorized users. You will be prompted to log in as soon as you access Resource Manager. If you are in the middle of work and need to step away from your desk, you can log out and log back in when you return using the **Log In** and **Log Out** links on the top right.

When logging in, you will be presented with a log in prompt.



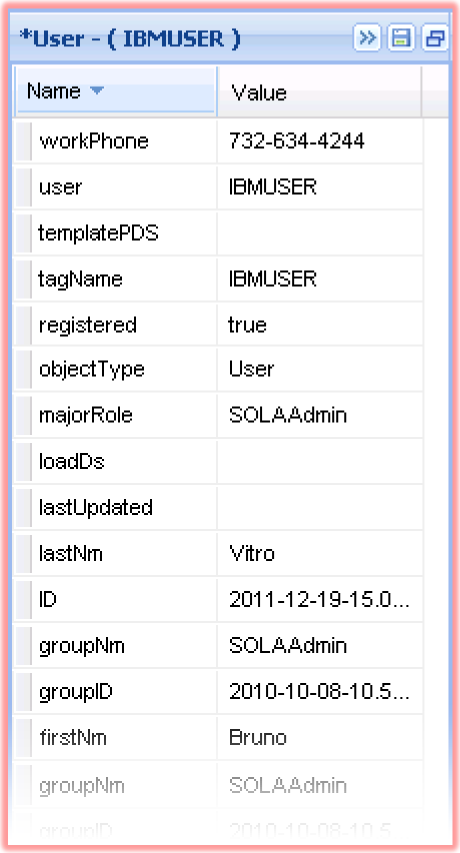
User SignOn

UserName: dbcrew

PassWord: •••••

Enter your RACF username and password and either press the Enter key or click the **SignOn** button. After you have logged in successfully, your username will be displayed above the **Log In / Log Out** links on the top right of the Resource Manager window.

Clicking on your name will display your user-level properties in the Properties panel.



*User - (IBMUSER)

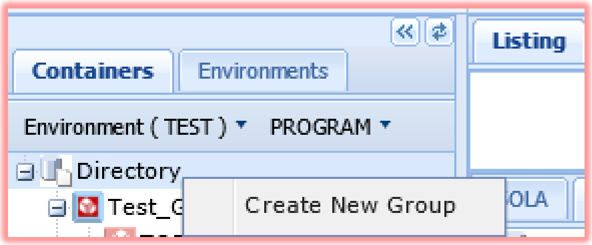
Name	Value
workPhone	732-634-4244
user	IBMUSER
templatePDS	
tagName	IBMUSER
registered	true
objectType	User
majorRole	SOLAAdmin
loadDs	
lastUpdated	
lastNm	Vitro
ID	2011-12-19-15.0...
groupNm	SOLAAdmin
groupID	2010-10-08-10.5...
firstNm	Bruno
groupNm	SOLAAdmin
groupID	2010-10-08-10.5

SOLA is highly customizable and so the properties displayed in this guide may not be what you see on your screen. Check with a SOLA Administrator if you have questions about specific properties and their values.

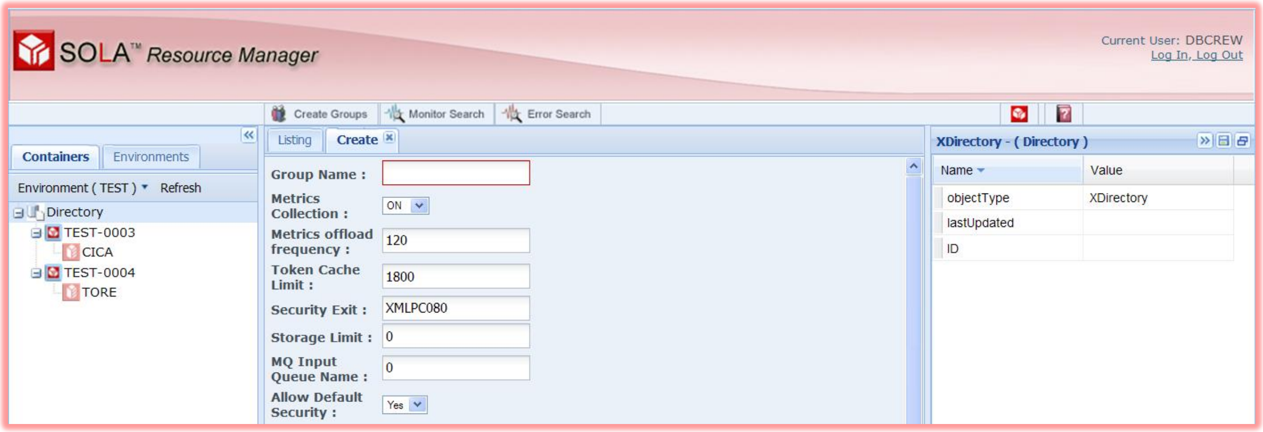
Users can change the values of some of their user properties by clicking on either an existing value or an empty field in the **Value** column. If the value is editable, a cursor will appear in its field. The value being edited can be either a text field or a drop down menu.

Working with Tabs

The Resource Manager workspace is tab based, which means that it can contain several active panels, each of which is represented by a tab. Right click on the Directory and click on Create New Group.



The illustration below will then show two active tabs in the workspace.



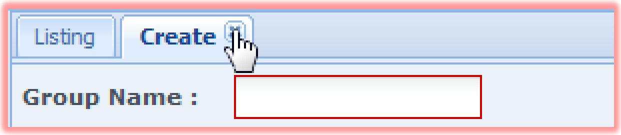
Group Name :
Metrics Collection : ON
Metrics offload frequency : 120
Token Cache Limit : 1800
Security Exit : XMLPC080
Storage Limit : 0
MQ Input Queue Name : 0
Allow Default Security : Yes

Name	Value
objectType	XDirectory
lastUpdated	
ID	

There is no set limit to how many tabs can be displayed at once; if too many tabs are open to fit into the current workspace size, a scroll bar will appear.

You can switch between active tabs at any time. This tab based functionality provides several useful benefits, such as the ability to stop work on something and come back to it later without having to start from scratch, and the ability to troubleshoot (error search, etc.) without having to abandon what you are working on.

To close a tab, click on the X button in the tab's top right corner.



Button Bar

The button bar provides shortcuts and access to some of Resource Manager's functions.

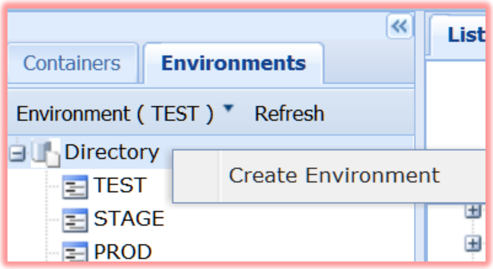
Click this button to access the group creation panel, which is used to create user groups.

Click this button to access the Monitor Search panel, which is used to search through the SOLA transaction log. See page 60 for details on how to use the transaction search panel.

Click this button to access the Error Search panel, which is used to search through the SOLA error log. See page 65 for details on how to use the error log.

Environment Panel

The Environment Panel defines the SOLA Group where Containers and Container Groups are defined.

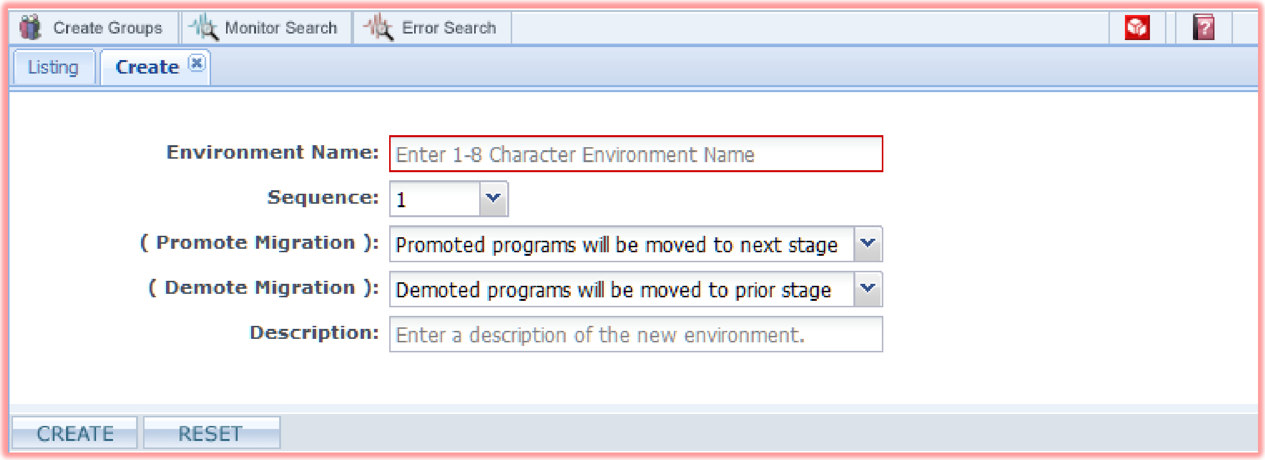


The Environment Panel displays a tree that functions in a manner very similar to that of Windows Explorer. Environments are represented by Directories, which act like Windows folders and define mainframe directories which hold containers (which are like files in folders). If a Directory has multiple Environments, that item can be expanded by clicking its associated "+" icon and collapsed by clicking the "-" icon.

Note: You will need to contact your SOLA Administrator for Environment Panel setup information.

Create Environment

The create environment panel lets you create SOLA environments that are then linked to backend environments using the promote.jcl file.



Environment Name: Enter 1-8 Character Environment Name

Sequence: 1

(Promote Migration): Promoted programs will be moved to next stage

(Demote Migration): Demoted programs will be moved to prior stage

Description: Enter a description of the new environment.

To create an environment, select an environment code, a sequence, promote and demote option, then enter a brief description such as test, production, etc.

- **Environment Name:** the environment name is a 1-8 character name that represents the environment. The name serves solely as an identifier for the environment, and typically is defined more closely matching your company naming conventions for TEST, STAGE or PROD.

- **Sequence:** the sequence represents the environment promotion hierarchy. The lower the sequence number, the lower the environment in the hierarchy. Typically, test environments occupy the lower rungs in the hierarchy, QA or stage environments somewhere in the middle and production environments occupy the highest rungs (and therefore would have the highest sequence numbers). It is recommended that you stagger your sequence numbers (e.g. 1,5 and 9 instead of 1,2 and 3) so that you will have room for additional environments. Sequence numbers do not have to be sequential (1,14, 58 is the same as 1,2,3).

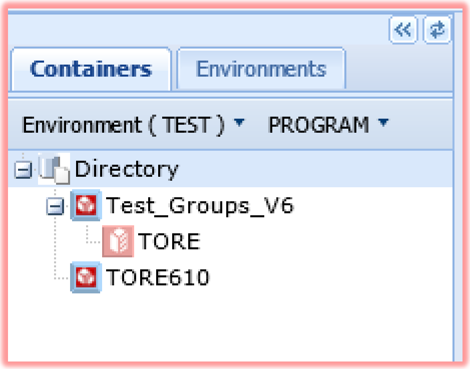
- **Promote Migration:** choose to 'Move' or 'Copy' programs to the 'next' stage.

- **Demote Migration:**   choose to 'Move' or 'Copy' programs to the 'prior' stage.

When you have made your selections, click CREATE to create the environment.

The Containers Panel

The Containers Panel displays a list of containers and container groups in a particular SOLA environment. The Containers Panel is the target for all administrative functions that can be performed in Resource Manager. Whenever accesses, policies, etc. are created and assigned to a program, they are not in effect until they are associated with a container or container group.



The Containers Panel displays a tree that functions in a manner very similar to that of Windows Explorer. Container groups are represented by container icons, which act like Windows folders and hold containers (which are like files in folders). If a tree item has members (again like files in a folder), that item can be expanded by clicking its associated "+" icon and collapsed by clicking the "-" icon.

The Containers Panel has a "Refresh" button ⟳ that will refresh the Directory contents of all Resource Manager panels.

Containers Panel Icons

Each item in the Containers Panel is represented by a distinct icon. The legend below shows a list of items displayed in the Containers Panel and their associated icons.

Directory root

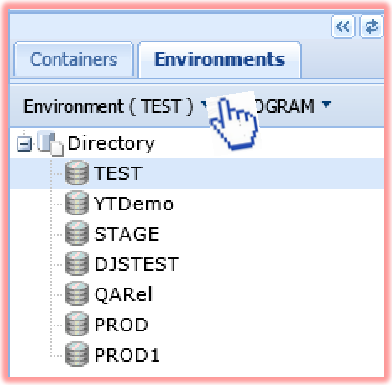
Container group

SOLA Container

Containers Panel Filters

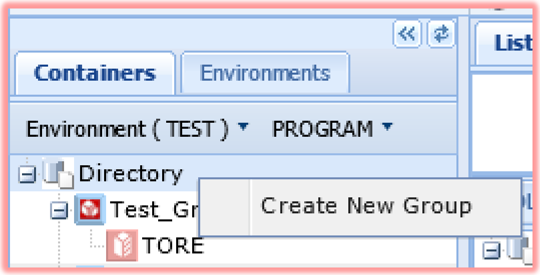
The contents of the Containers Panel can be filtered by environment. Environment names are customizable and so the names displayed here may not be what you see on your screen.

Select the desired environment from the **Environments** menu. Only containers and container groups belonging to that environment will be displayed in the tree. For example, if you select the TEST environment, only containers and container groups in the TEST environment will appear in the Containers Panel.



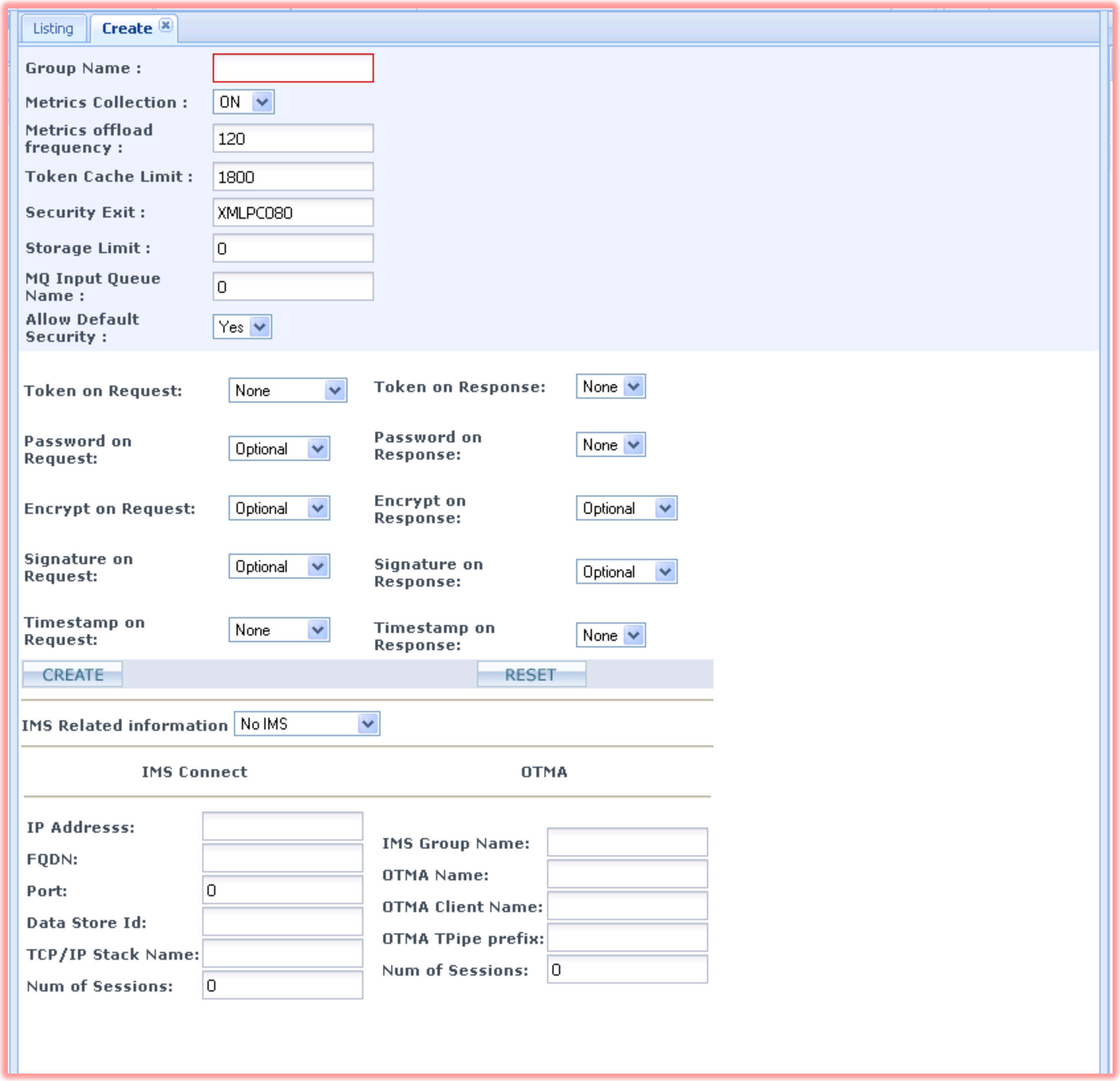
Container Panel Menus

Right clicking on an item in the containers panel will display a pop-up menu.



Directory Menu

Create New Group: displays the Create tab in the workspace, allowing you to create a new container group. Creating container groups is detailed on page 25.



Group Name :

Metrics Collection : ON

Metrics offload frequency : 120

Token Cache Limit : 1800

Security Exit : XMLPC080

Storage Limit : 0

MQ Input Queue Name : 0

Allow Default Security : Yes

Token on Request:	None	Token on Response:	None
Password on Request:	Optional	Password on Response:	None
Encrypt on Request:	Optional	Encrypt on Response:	Optional
Signature on Request:	Optional	Signature on Response:	Optional
Timestamp on Request:	None	Timestamp on Response:	None

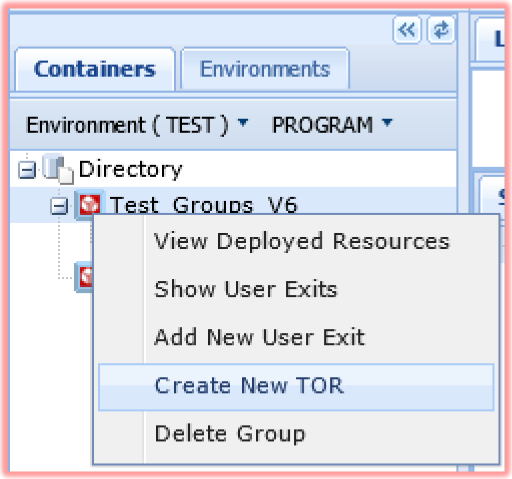
IMS Related information No IMS

IMS Connect		OTMA	
IP Addresss:		IMS Group Name:	
FQDN:		OTMA Name:	
Port:	0	OTMA Client Name:	
Data Store Id:		OTMA TPipe prefix:	
TCP/IP Stack Name:		Num of Sessions:	0
Num of Sessions:	0		

Container Group Menu

A container is an environment in which the SOLA runtime is deployed. When working with CICS, the SOLA runtime container runs in a TOR (Terminal Owning Region). Resource Manager gives you the ability to use an existing TOR region as a SOLA container by registering that region to SOLA.

Note: You can collect containers into groups to be managed as a single entity.



Create New TOR: displays the Create tab in the workspace, allowing you to create a new TOR region in the selected group. Creating regions is detailed on page 34.

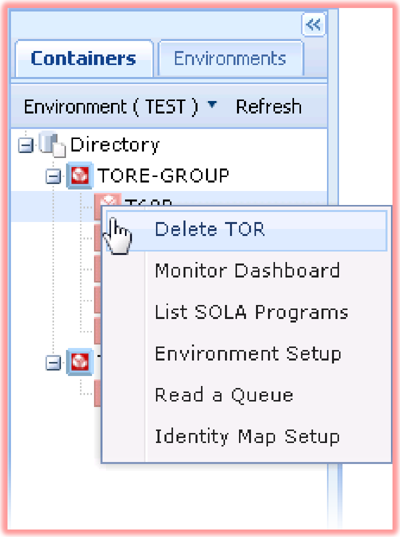
Show User Exits: displays the User Exits defined for the Container Group. You can delete exits by right clicking on an exit name and selecting **Delete Exit** from the pop up menu.

Add New User Exit: displays the Create panel which allows you to add a new User Exit to the Container Group. Specify an exit name, type (options are SOAP Header Input, SOAP Header Output and Status) and Optional description.

View Deployed Resources: displays the Deployed Manager panel in the workspace. This tab displays the Deployed IPA tab which shows those IP Addresses that have been deployed in the selected container group (nothing takes effect in Resource Manager until it is associated with/deployed in a container group).

Delete Group: deletes the container group and all containers in the group.

Container Menu



Delete TOR: deletes the selected container.

Monitor Dashboard: displays the monitoring dashboard for the selected container.

List SOLA Programs: displays a window containing information on SOLA internal programs running in that container. This is for debugging purposes. Information on each program is organized under a series of columns which can be sorted in ascending or descending order:

- **RowId:** a sequence number with no significance outside of identifying the program's location in the table.

- **Program:** the name of the SOLA internal program.
- **PTF:** the latest PTF (Program Temporary Fix) applied to the program.
- **Length:** the length of the load module.
- **Lang:** the language the program is written in. Options are A for Assembler and L for Language Environment.
- **Date:** the day the program was compiled.
- **Time:** the time the program was compiled.

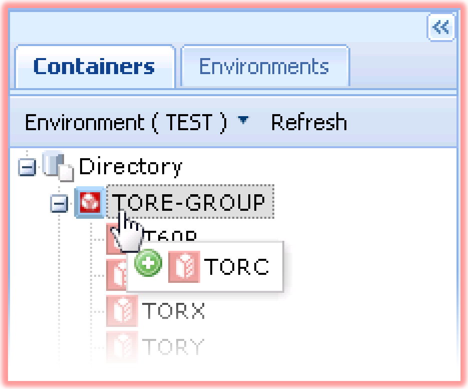
Environment Setup: displays a window containing information on the container's environment setup. The information is organized under a series of columns:

- **RowId:** a sequence number with no significance outside of identifying an item's location in the table.
- **CICS Release:** the version of CICS (if applicable).
- **CBRF Term:** the name of the CBRF terminal (for 3270 Linkable Bridge support) (if applicable).
- **BRNSF DS:** the dataset name of the BRNSF dataset (if applicable).
- **BRCV:** yes or no (if applicable).
- **BRMP:** yes or no (if applicable).
- **DFH3270:** whether the linkable bridge is installed (yes or no) (if applicable).
- **CSKL:** whether the socket listener is installed (if applicable).
- **RANDOM:** whether ICSF is active in the container.

Read a Queue: allows you to read a CICS TSQueue. This is useful for debugging, specifically for reading the SOLA trace queue, SOLATRACE. With SOLA IMS Container, trace records are written to SYSOUT.

Identity Map Setup: displays the Identity Map Setup panel. See the **Identity** section on page 71 for information on how to use this functionality.

Container Panel Drag and Drop

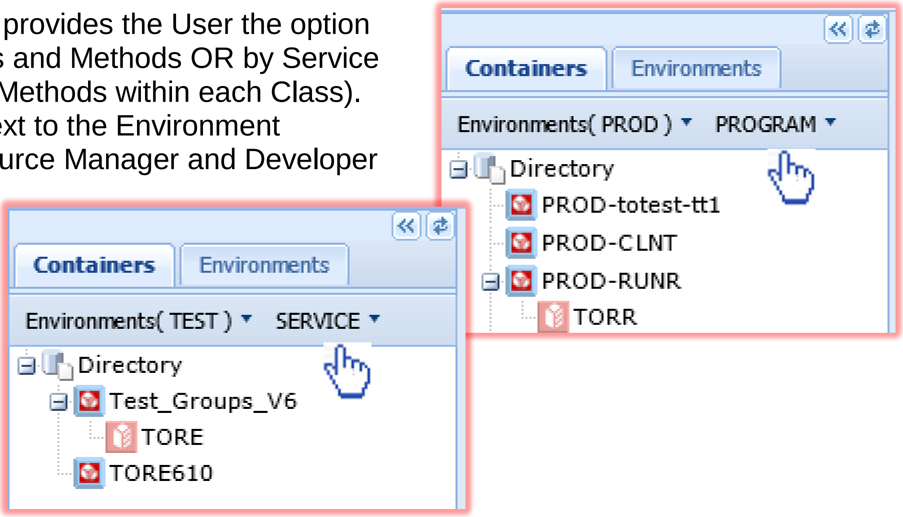


You can drag SOLA containers from one group to another by dragging a container and dropping it onto the target group.

Items from other trees can be dragged to other SOLA container groups to be deployed in that container group. These operations are discussed in detail in the Using Resource Manager section on page 24.

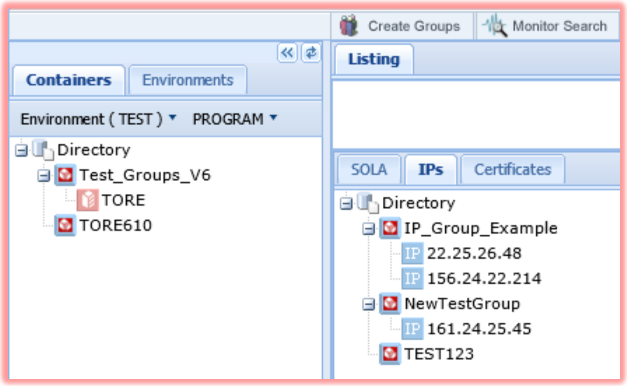
Program and Service View

Program and **Service** view provides the User the option to view objects by Programs and Methods OR by Service and Operations (Class and Methods within each Class). This dropdown is located next to the Environment dropdown on both the Resource Manager and Developer home pages.



Resource Panel

The Resource Panel is a tabbed panel that displays one of three trees: **SOLA** projects and programs, **IPs** groups and addresses, or **Certificates**.



SOLA Tab

The SOLA tab displays the SOLA directory. This is the same information displayed in the directory tree in SOLA Developer and is filtered by environment as selected in the Containers Panel. This tab is used to assign user access rights and polices to programs or methods while
in **Program** view.

IPs Tab

The IPs tab displays a list of IP groups and associated IP addresses. SOLA incorporates an IP filtering component, which is useful for those companies that aren't ready to move to using WS-Security for access control. The IPs tab is used to manage IP filtering, and it displays a list of IP addresses and IP address groups. This is discussed in greater detail on page 43.

Certificates Tab

The Certificates tab is used to keep track of certificates that are uploaded to SOLA. Using certificates is detailed on page 35.

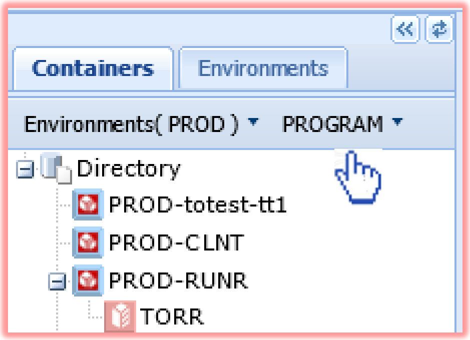
Resource Panel Icons

Each item in the resource panel is represented by a distinct icon. The legend below shows a list of containers panel items and their associated icons.

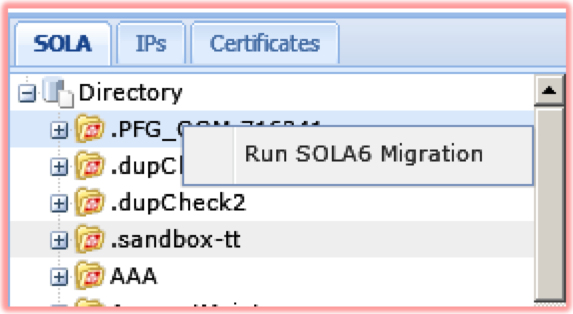
	Directory root or		Adhoc SQL program
	Project		Custom program
C	Commarea program		Stored Procedure program
	Containers program		BPEL program
☎	Callable API program		Certificate
⬆	Outbound Commarea program		Mask Group – deprecated
B	BMS3270 program		Program Mask - deprecated
I	IMS Program		

SOLA Tab Menus

This section describes the functionality of the SOLA Tab Menus when in **Program** view only, as seen in the illustration on the right:

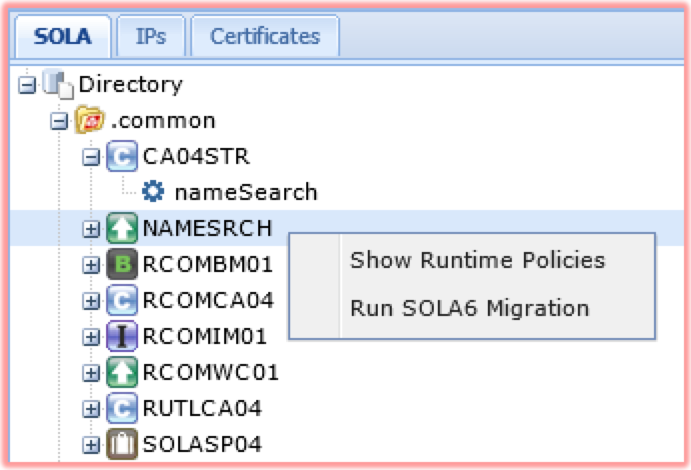


Project Context Menu



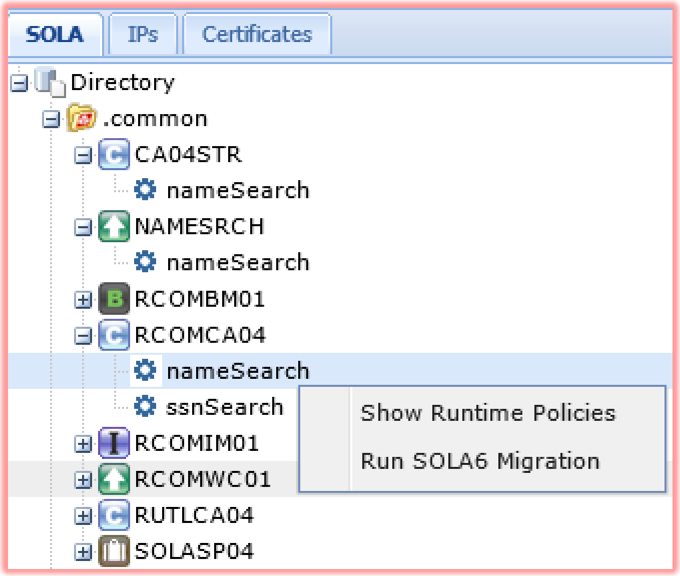
Run SOLA 6 Migration: this option is for SOLA 5.x customers who plan to migrate to SOLA 6.x and have copied their SOLA 5.x file system to a SOLA 6.x environment. Selecting this option will trigger an update of the SOLA 6.x database with metadata from the SOLA 5.x filesystem.

Program Context Menu



Show Runtime Policies: displays all of the policies assigned to the program in a new tab. You can remove a policy group (and all the policies it contains) by right clicking the group name and selecting **Remove Policy** from the menu.

Note: SOLA Developer Studio offers a new Policy Management option as a sub-option under Program context. This feature is lot more intuitive in terms of managing/administering policies



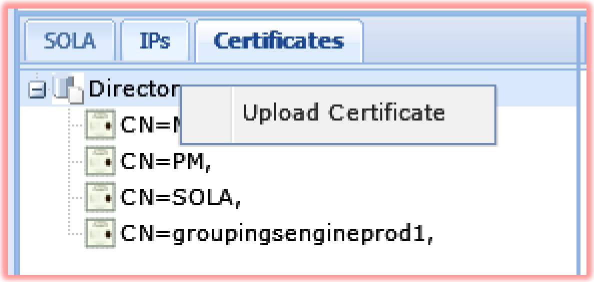
Method Context Menu

The method menu contains the same choices as the program menu, but displays the information for that particular method only.

NOTE: accesses and policies assigned to the method's parent program will apply to the method even though they will not display when viewing method specific information. Only those accesses and policies specifically applied to the method will be displayed.

Certificates Tab Menus

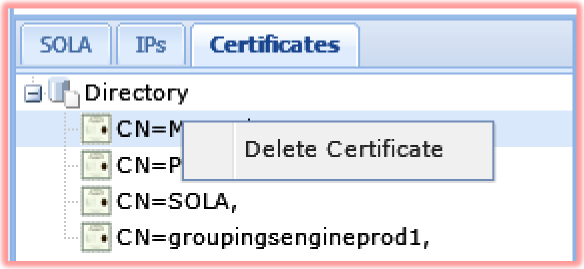
Directory Menu



Upload Certificate: displays the Upload Cert tab which allows you to upload a certificate to Resource Manager. Uploading certificates is detailed on page 39. Resource Manager does not allow you to download a certificate that has been uploaded.

NOTE: if you are using Websphere 6.1, then you must convert your certificates from binary to Base64 encoded prior to uploading.

Certificate Menu

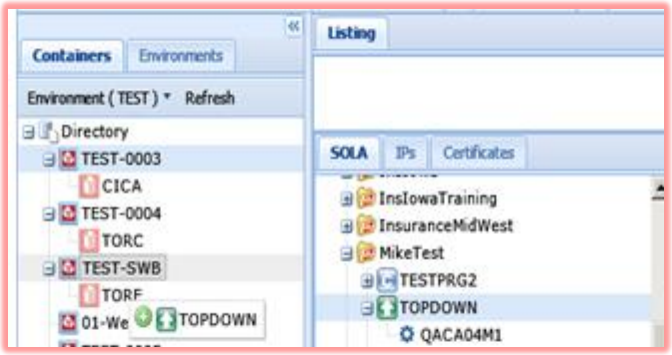


Delete Certificate: deletes the certificate.

Resource Panel Drag and Drop

All of the trees in the resource panel have drag and drop capabilities. All of the drag and drop operations that involve the containers panel are discussed in greater detail in the section Using Resource Manager, which starts on page 24.

When moving items, keep the following rules in mind:



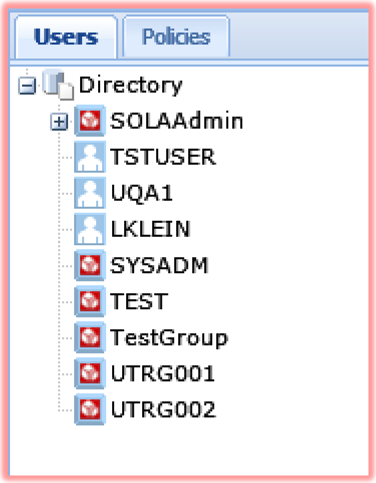
Programs: a program can be moved to the Containers panel to have its accesses and policies deployed in a container group. You can only drag a program to a container group itself. Dragging the program to a container group will deploy that program, along with its accesses and policies to every container in that container's group.

Methods: methods can be deployed to a container group (same rules as programs) to deploy that method (and its accesses and policies) to all the containers in that group.

IPs: Deployment of IP groups for IP filtering is enabled by dragging the IP Group directly to the Runtime container group. Please refer Administration Guide "APPENDIX B: SOLA CUSTOM CHANNEL LOCKDOWN SECURITY" for more information on how to setup IP Filtering.

Certificates: certificates can be deployed to container groups (same rules as programs).

Subjects Panel



Users Policies

- Directory
  - SOLAAdmin
  - TSTUSER
  - UQA1
  - LKLEIN
  - SYSADM
  - TEST
  - TestGroup
  - UTRG001
  - UTRG002

The subjects panel is a tabbed panel that displays one of two trees; Users and Policies.

This panel is used to create and work with the subjects of Resource Manager; subjects are assigned to resources (programs or methods, IP addresses/groups or certificates) creating either a User or an assigned Policy association with the Resource. These items are then deployed to container groups.

Users Tab

The Users tab displays a list of all SOLA users and user groups. Dragging a user or user group to a container group creates an association. That association is then considered deployed to the container group it was dragged onto. This is discussed in greater detail on page 40.

Policies Tab

The policies tab contains a list of policies, represented by groups and group members. The group is the actual policy while the group members are aspects of the policy. This is discussed in greater detail on page 52.

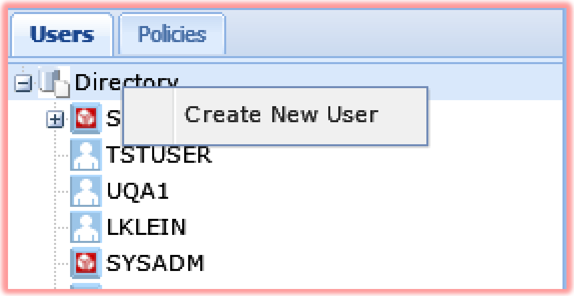
Subjects Panel Icons

Each item in the resource panel is represented by a distinct icon. The legend below shows a list of containers panel items and their associated icons.

🗂	Directory root
🟥	User group or Policy (not group)
👤	User
□	Policy aspect (detail)

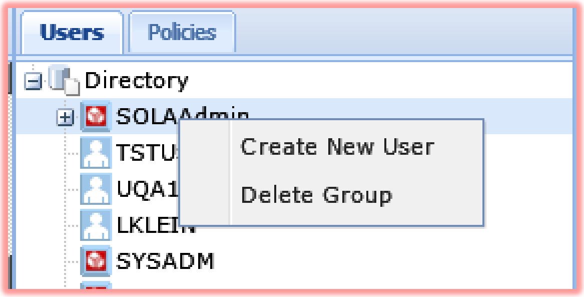
Subjects Panel Menus

All of the tabs of the subjects panel feature the same menus, except for the policy tab which does not allow you to create policies within a policy group. This is because the policy group is the actual policy, and the policy items in that group are aspects of that policy.



Directory Menu

Create New User or Policy: displays the Create tab, allowing you to create a new subject (user or policy, depending on which tab you are working in).

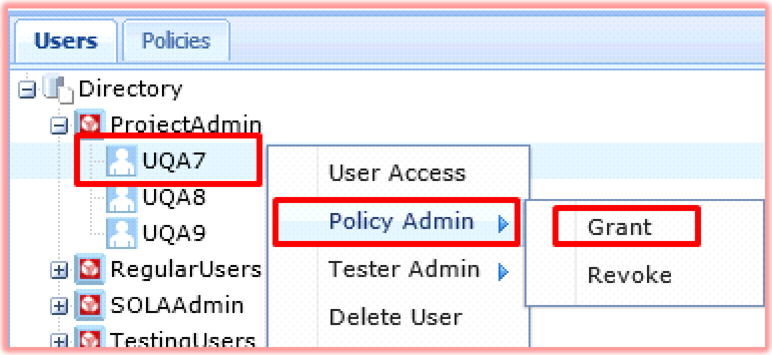


Group Menu

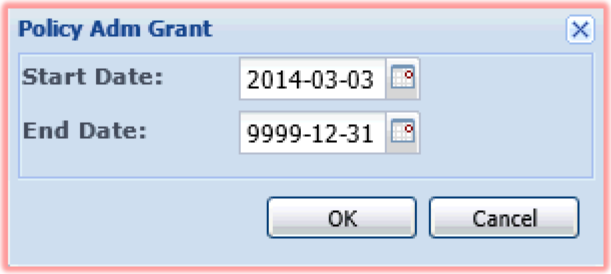
Create New User: displays the Create tab, allowing you to create a new subject (user or policy depending on which tab you are working in). Creating subjects is discussed in multiple sections (based on subject type).

Delete Group: deletes the user group and deletes (or orphans where appropriate) all subjects (users or policies) in that group.

Subject Menu



Policy Admin: will allow the SOLA Administrator the ability to Grant or Revoke 'Policy Admin' rights to a Project Administrator. The Project Administrator will then have the authority to adjust policies for programs or methods using 'Policy Management' sub-option under Program context in SOLA Developer Studio. After clicking on 'Grant' a dialog box will appear where SOLA administrator can set the validity of the authority by setting Start/End Dates as illustrated below

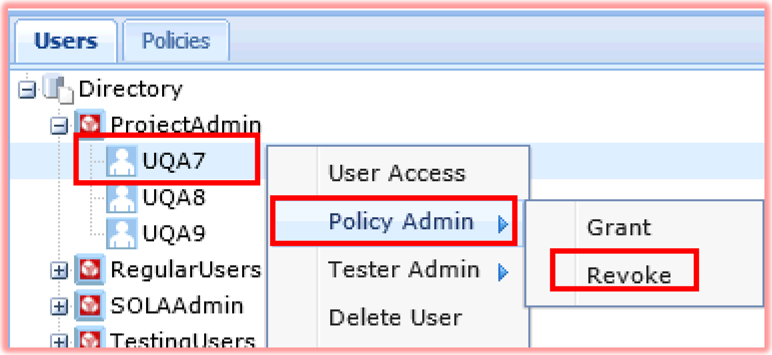

Policy Adm Grant

Start Date: 2014-03-03

End Date: 9999-12-31

The Start and End dates are both inclusive dates for checking the authority. Once the current date is past 'End-Date', the authority granted is automatically expired.

Policy Admin access can be either left to auto-expire based on appropriate setting on the 'End Date' or SOLA Administrator can explicitly revoke the authority by clicking on Revoke option as illustrated below.

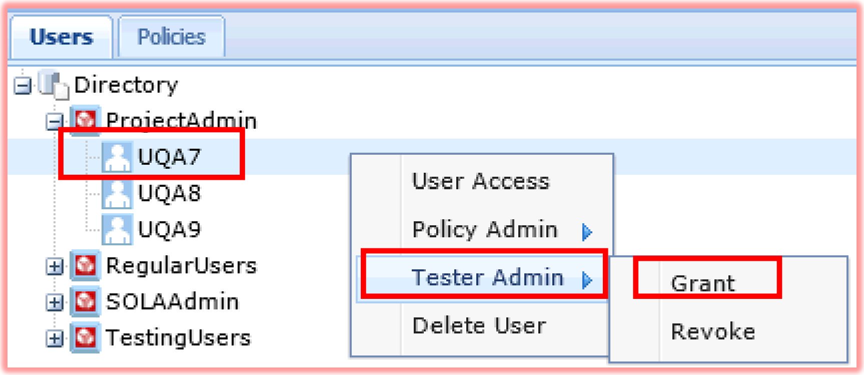


Note: Policy Admin Authority is only available for users who are defined as Project Administrators of one or more projects. By granting this access to Project Administrators, it will allow them to administer policies for programs under the respective projects to which they are defined as administrators. By granting Policy Admin Authority to a regular user does not activate any special privileges.

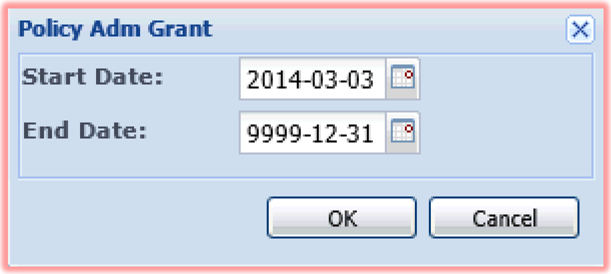
Tester Admin: will allow the SOLA Administrator the ability to Grant or Revoke 'Tester Admin' rights to a SOLA user. Tester Admin authority is used to control general access to 'Raw Soap Test' screen as well as access to restricted end-points under quick test. Following Matrix defines the privileges that are inherited by the user based on the role of the user

Project Admin	Tester Admin	Quick Test		Raw Soap-Test	
		Open End-points	Restricted End-points	Open End-points	Restricted End-points
Y	Y	Y	Y	Y	Y
Y	N	Y	N	N	N
N	Y	Y	N	Y	Y
N	N	Y	N	N	N

The following illustration shows how to Grant 'Tester Admin' to a user.



After clicking on 'Grant' a dialog box will appear where SOLA administrator can set the validity of the authority by setting Start/End Dates as illustrated below

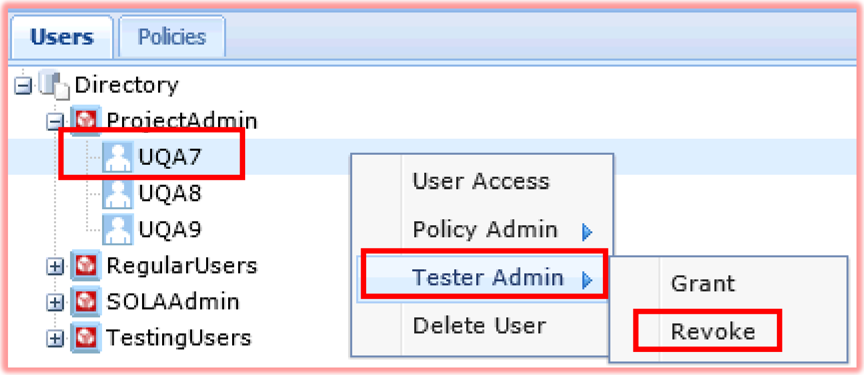

Policy Adm Grant

Start Date: 2014-03-03

End Date: 9999-12-31

The Start and End dates are both inclusive dates for checking the authority. Once the current date is past 'End-Date', the authority granted is automatically expired.

A Tester Admin access can be either left to auto-expire based on appropriate setting on the 'End Date' or SOLA Administrator can explicitly revoke the authority by clicking on Revoke option as illustrated below



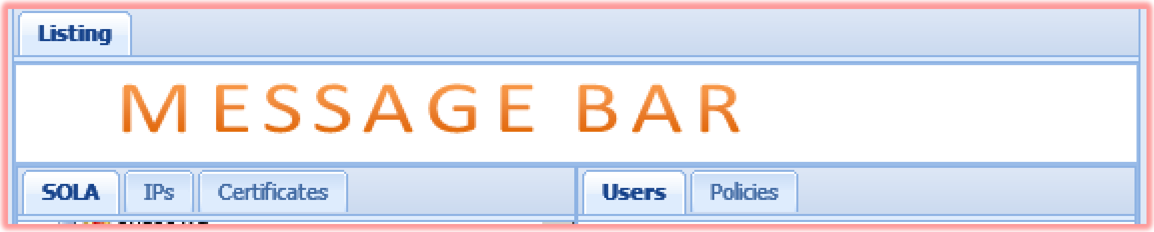
Delete User: deletes the User.

Subjects Panel Drag and Drop

Some subjects can be dragged and dropped into or out of groups. Users can exist outside groups and can be dragged into the root directory to be removed from a group. Policy aspects can only exist within a group and cannot be moved from policy to policy.

Subjects from the subjects panel, i.e. User Groups or Policies can be dragged to resources (except certificates) in the resources panel. Dragging a user group into a resource creates an access (that subject is authorized to access that resource). Dragging a policy into a resource applies that policy to that resource (e.g. the resource is now bound by that policy).

Message Bar



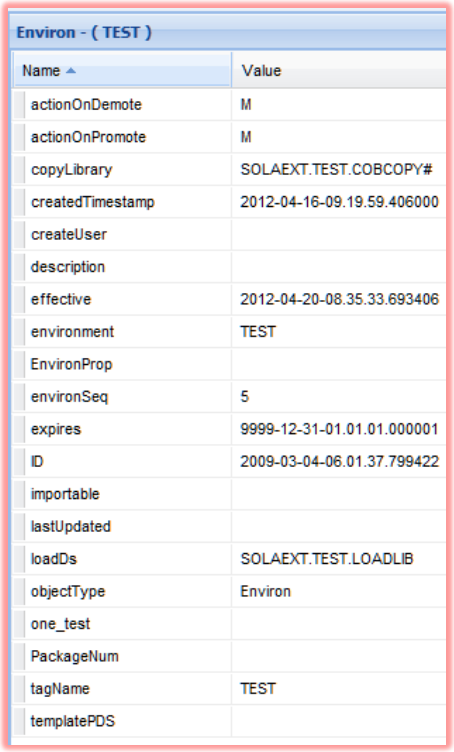
The message bar is used to display Resource Manager system messages.

Properties Panel

The properties panel is on the right of the screen. Properties differ, depending on the object that's selected. The property window below shows the default properties for a Project.

Properties are extensible. You can add custom values to objects by modifying the schema for the property. Adding and modifying property schemas is documented in section "Custom Schema" in the SOLA User's Guide.

.



Environ - (TEST)

Name	Value
actionOnDemote	M
actionOnPromote	M
copyLibrary	SOLAEXT.TEST.COBCOPY#
createdTimestamp	2012-04-16-09.19.59.406000
createUser	
description	
effective	2012-04-20-08.35.33.693406
environment	TEST
EnvironProp	
environSeq	5
expires	9999-12-31-01.01.01.000001
ID	2009-03-04-06.01.37.799422
importable	
lastUpdated	
loadDs	SOLAEXT.TEST.LOADLIB
objectType	Environ
one_test	
PackageNum	
tagName	TEST
templatePDS	

Using Resource Manager

This section outlines the use of Resource Manager. It is recommended that you familiarize yourself with the basics of Resource Manager in the previous section before continuing onto this section.

Resource Manager Made Simple

Resource Manager is a very powerful run-time governance tool. The essence of its use is actually quite simple, and can be summarized in a few short paragraphs. Most users will find the following brief descriptions of how to use the application sufficient to begin using it right away. For those who require more information, or are unfamiliar with some of the concepts used in Resource Manager, more detailed instructions are provided later in this section.

Users

The most basic use of Resource Manager is to create and configure a User. Once you've created your user in the Subject panel, drag the User Group containing the newly created user from the subjects panel into a resource in the resources panel to create an "access." Once you have an access established (a User that applies to a method, program or group of programs), you can drag that access to the containers panel to deploy that access for a container group.

IP Addresses

IP addresses are maintained within the IP address group (now located in the Resource panel). Once IP addresses have been created or deleted and the IP Group has been dragged directly into a Container group (which will apply all IP addresses in that group to all containers in the container group) a panel is presented that provides you with an opportunity to review the IP addresses that have been inserted/added or deleted, along with all other IP addresses that remain untouched in the group.

Policies

You can use Resource Manager to create and configure policies, then drag those policies to a resource in the resources panel, creating an assigned policy. Applicable resources are either programs or methods in the SOLA Directory.

Once you have an assigned policy, you can drag the Policy group to the containers panel to put it into effect for a Container group.

Certificates

You can upload a trusted certificate from a SAML authority. Doing so will allow Resource Manager to authenticate that authority's tokens. You can also upload untrusted certificates to verify application server credentials for servers that invoke SOLA web services.

Working with the Containers Panel

The first step in using Resource Manager is the creation of container groups and containers. In the CICS version of SOLA, containers represent actual CICS TOR regions on the mainframe that SOLA will need to interact with, while container groups represent groupings of those containers.

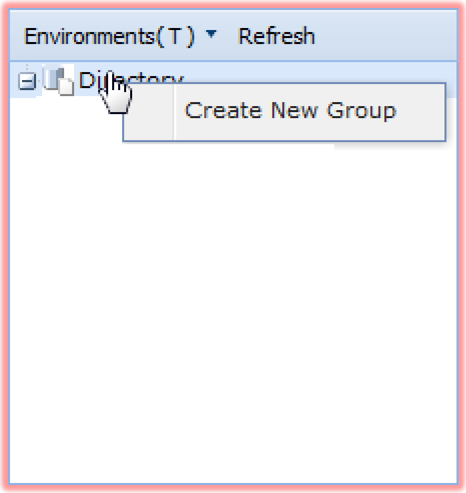
When you first launch Resource Manager, the Containers panel will be empty. It will be up to you to populate it with all of the SOLA containers you will need to use with SOLA and to organize those containers into groups within the Runtime database.

Container Groups

Container groups are more than just containers for groups; they also serve to control metrics collection and security policies for the containers inside them. Groups allow you to do the following:

- Enable and configure metrics collection
- Enable use of the default security policy
- Designate a security user exit
- Configure cache, queue and storage options
- Enable and configure custom security policy

Creating Container Groups

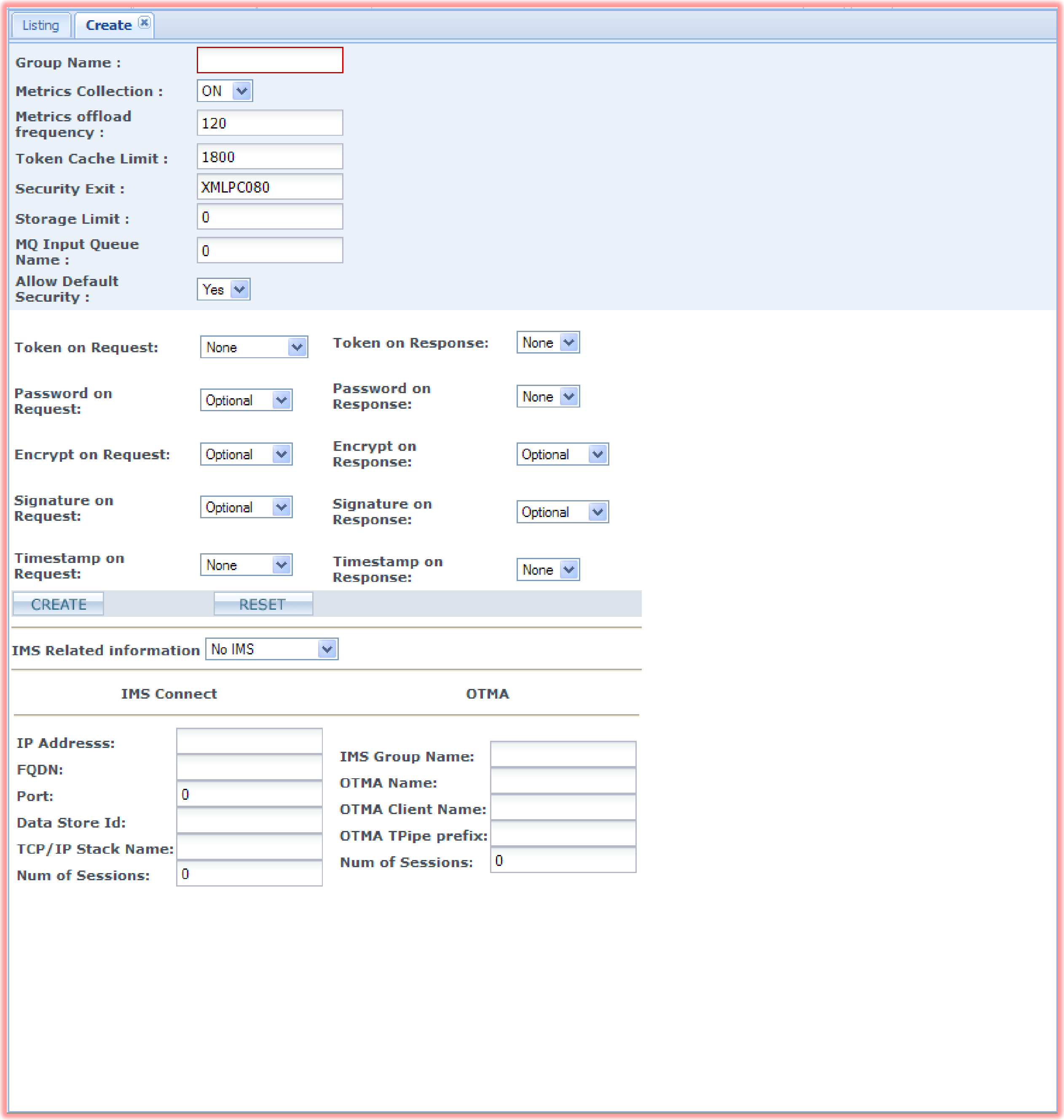


All containers in Resource Manager must be contained in a container group. To create a container group, right click on the directory icon and select **Create New Group** from the menu.

This will display the Create tab in the workspace, allowing you to create a new container group within the Runtime database.

We recommended that you group your containers based on their security level (low security group for test containers, high security group for production containers, etc.). This will make assigning access a lot simpler.

The create tab contains a series of fields that you will need to populate to create a new group. All fields/menus except for the group name are preconfigured with the default settings.



Group Name :	
Metrics Collection :	ON
Metrics offload frequency :	120
Token Cache Limit :	1800
Security Exit :	XMLPC080
Storage Limit :	0
MQ Input Queue Name :	0
Allow Default Security :	Yes

Token on Request:	None	Token on Response:	None
Password on Request:	Optional	Password on Response:	None
Encrypt on Request:	Optional	Encrypt on Response:	Optional
Signature on Request:	Optional	Signature on Response:	Optional
Timestamp on Request:	None	Timestamp on Response:	None

IMS Related information: No IMS

IMS Connect		OTMA	
IP Addresss:		IMS Group Name:	
FQDN:		OTMA Name:	
Port:	0	OTMA Client Name:	
Data Store Id:		OTMA TPipe prefix:	
TCP/IP Stack Name:		Num of Sessions:	0
Num of Sessions:	0		

To create a group with the default settings, fill in the **Group Name** field and click **CREATE**. To configure custom settings for the group, you will need to make changes to the following settings.

Standard Settings:

- **Metrics Collection:** enables (ON) or disables (OFF) metrics collection (by SOLA) for the containers in the group.

- **Metrics offload frequency:** determines how often, in seconds, metrics are spooled to the database.

- **Token Cache Limit:** How long, in seconds, before a cached token expires. The lowest limit is 10 seconds and anything below 10 will reset it to 1200 seconds. After updating this value, wait 3 minutes for it to take effect.

- **Security Exit:** specifies the program to be used as security exit. By default, XMLPC080, the SOLA security exit, is used

- **Storage Limit:** the maximum size of an outbound message

- **MQ Input Queue Name:** the name of the MQ queue that SOLA will listen to for input MQ messages

- **Allow Default Security:** specifies whether the containers in the group will use the default security policy. Choosing "No" will force the containers in the group to use the custom security policy, defined below

Security Policy Settings: These settings create a default policy for the Container Group.

- **Token on Request:** this setting determines whether SOLA will accept requests without an attached security token.
  - **NO:** SOLA will allow requests without security tokens
  - **MainframeID:** SOLA will require a mainframe user id as a security token.
  - **LDAP ID:** SOLA will require an LDAP user id as a security token.
  - **SAML:** SOLA will require SAML credentials as a security token.
  - **Restrict by IP:** whether only certain IP addresses can submit requests

- **Token on Response:** with the current version of SOLA, the only option is NO.

- **Password on Request:** this setting determines whether SOLA accepts requests that have a token, but no password
  - **Optional:** a password is not required (SOLA will accept requests without a password).
  - **Mandatory:** a password is required.

- **Password on Response:** with the current version of SOLA, the only option is NO.

- **Encrypt on Request:** this setting determines whether SOLA accepts requests that are not encrypted.
  - **Optional:** encryption is not required (SOLA will accept requests without encryption).

  - **Mandatory:** encryption is required.

- **Encrypt on Response:** this setting determines whether SOLA will encrypt responses
  - **Optional:** SOLA will not encrypt responses
  - **Mandatory:** SOLA will encrypt responses

- **Signature on Request:** this setting determines whether SOLA accepts requests without an attached signature.
  - **Optional:** attached signatures are not required (SOLA will accept requests without attached signatures).
  - **Mandatory:** the body of the SOAP request must be signed.

- **Signature on Response:** this setting determines whether SOLA will attach a signature to responses.
  - **Optional:** SOLA will not attach a signature to responses.
  - **Mandatory:** SOLA will attach a signature to responses.

- **Timestamp on Request:** this setting determines whether SOLA accepts requests without an attached timestamp. The timestamp contains the policy's expiration date and time.
  - **None:** attached timestamps are not required (SOLA will accept requests without attached timestamps).
  - **Mandatory:** attached timestamps are required.

- **Timestamp on Response:** with the current version of SOLA, the only option is NO.

IMS Related Information:

If you're planning to use this SOLA Container to service enable IMS transactions then choose either **USE IMS Connect** or **Use OTMA** from the dropdown, otherwise leave it at the default **No IMS**. If you choose **USE IMS Connect** then fill in the values in the left column, otherwise with **Use OTMA** fill in the values in the right column.

- **IMS Connect Values:**
  - **IP Address:** Enter either the IP Address of the TCPIP stack that IMS Connect connects to, or the FQDN below.
  - **FQDN:** The Fully Qualified Domain Name of the TCPIP stack that IMS Connect connects to.
  - **Port:** The Port that IMS Connect is listening on.
  - **Data Store Id:** The Datastore name corresponding to the configuration parameter "ID" in the "DATASTORE" configuration statement of IMS Connect.
  - **TCP/IP Stack Name:** The name of the TCPIP stack that IMS Connect connects to.

   - **Num of Sessions:** Max number of connections enabled with IMS Connect. This parameter is ignored with this release.
- **OTMA Values:**
   - **IMS Group Name:** IMS XCF Group name as defined by parm GRNAME in the IMS subsystem.
   - **OTMA Name:** The name that IMS will have within the XCF group (specified by parm OTMNAM in the IMS subsystem).
   - **OTMA Client Name:** Unique user defined name with which the OTMA client will be defined in the OTMA connection (8 characters).
   - **OTMA TPipe Prefix:** 4 character prefix for the Transaction Pipe enabled for the OTMA sessions.
   - **Num of Sessions:** Max number of concurrent sessions enabled with the OTMA connection.

When you are finished configuring the group, click **CREATE**. You can reset all the settings to their defaults at any time clicking the **RESET** button.

After creating the group, there may be additional values that are required for IMS Connect. You specify those values by modifying the group, as described below.

Modifying Container Groups

Container Group values are modified in the Properties pane.

To add IMS access to an existing container, or to modify the IMS settings, you modify the following property values for the container group:

- **imsUsage:** To add IMS access to a container, modify this value from the default space by choosing either **C** or **O** from the dropdown. **C** indicates that you intend to use **IMS Connect,** and **O** indicates **OTMA**.

- **IMS Connect Values:**
   - **IMSCCommitMode:** The CommitMode value '0' determines whether the IMS Transaction is committed either before sending back the output message to the OTMA Client **OR** CommitMode value '1' determines if Output segments are first returned to the OTMA client and then the IMS transaction is committed.

     Note: See Appendix C: of the SOLA Developer Users Guide for additional information and examples of the IMSCCommitMode property.
   - **IMSCDataStoreId:** The Datastore name corresponding to the configuration parameter "ID" in the "DATASTORE" configuration statement of IMS Connect.
   - **IMSCfqdn:** The Fully Qualified Domain Name of the TCPIP stack that IMS Connect connects to.
   - **IMSCIPAddress:** Enter either the IP Address of the TCPIP stack that IMS Connect connects to, or the FQDN below.
   - **IMSCPort:** The Port that IMS Connect is listening on.

- **IMSCSyncLevel:** The synchronization level specifies the level of acknowledgement for each transaction. If Transactions are specified with **Synch Level=none**, no acknowledgement is required from the client. This option is not a valid option for transactions specified with CommitMode 0. If Transaction are specified with **Synch Level=confirm**, the client is required to send an acknowledgement to signal to IMS Connect whether or not the output message was successfully (ACK) or unsuccessfully (NAK) processed by the client.

  - If IMSCSynchLevel=confirm is requested with CommitMode 0, and the client responds with ACK, the transaction processing is completed. If the client responds with NAK, the output message will be requeued in IMS for later delivery.

  - If IMSCSynchLevel=confirm is requested with CommitMode 1, and the client responds with ACK, the database changes are committed. If the client responds with NAK, the database changes are backed out and the output message is discarded by IMS.

- **IMSCTCPip:** The name of the TCPIP stack that IMS Connect connects to.
- **IMSCIRMMsgId:** The name of the IMS Connect exit to be invoked. If not specified then SOLA will use the value *IRMREQ* when building the IRM Header, unless a value for IMSCIRMMsgId is specified on the soap:Header. If SOLA uses *IRMREQ* then the default IMS connect exit HWSIMSO0 is used. See the SOLA User's Guide for information on overriding IMS Connect values on the soap:Header.
- **IMSCIRMHDRClientID:** The client name to be passed to IMS Connect. If not specified, and if no value for IMSCIRMMHDRClientID is specified on the soap:Header, then IMSCIRMMHDRClientID is constructed by concatenating an 'M' or an 'S' ('M' for Main Program, 'S' for Subroutine) with the task number. See the SOLA User's Guide for information on overriding IMS Connect values on the soap:Header.
- **IMSCIRMHDRTermID:** The terminal ID to be passed to IMS Connect. If not specified then no terminal ID will be passed to IMS Connect, unless a value for IMSCIRMMHDRTermID is specified on the soap:Header. See the SOLA User's Guide for information on overriding IMS Connect values on the soap:Header.
- **IMSCNumSessions:** Max number of connections enabled with IMS Connect. This parameter is ignored with this release.

- **OTMA Values:**
  - **ImsGroupNm:** IMS XCF Group name as defined by parm GRNAME in the IMS subsystem.
  - **ImsNumSessions:** Max number of concurrent sessions enabled with the OTMA connection.
  - **ImsOTMAClientNm:** Unique user defined name with which the OTMA client will be defined in the OTMA connection (8 characters).
  - **ImsOtmaNm:** The name that IMS will have within the XCF group (specified by parm OTMNAM in the IMS subsystem).

- **ImsOtmaTPipePfx:** 4 character prefix for the Transaction Pipe enabled for the OTMA sessions.

To modify the Default Policy for an existing container you would modify the following property values:

- **InPassReqd:** this setting determines whether SOLA accepts requests that have a token, but no password
  - **N:** a password is not required (SOLA will accept requests without a password).
  - **Y:** a password is required.
- **InputEncrType:** this setting determines whether SOLA accepts requests that are not encrypted.
  - **N:** encryption is not required (SOLA will accept requests without encryption).
  - **R3:** RSA 3DES encryption is required.
  - **R1:** RSA DES encryption is required.
- **InSignatureReqd:** this setting determines whether SOLA accepts requests without an attached signature.
  - **N:** attached signatures are not required (SOLA will accept requests without attached signatures).
  - **Y:** the body of the SOAP request must be signed.
- **InTimestampReqd:** this setting determines whether SOLA accepts requests without an attached timestamp. The timestamp contains the policy's expiration date and time.
  - **N:** attached timestamps are not required (SOLA will accept requests without attached timestamps).
  - **Y:** attached timestamps are required.
- **InTokenReqd:** this setting determines whether SOLA will accept requests without an attached security token.
  - **N:** SOLA will allow requests without security tokens
  - **M:** SOLA will require a mainframe user id as a security token.
  - **L:** SOLA will require an LDAP user id as a security token.
  - **C:** SOLA will require a Custom Security Token as a security token.
  - **I:** Enables IP Filteirng
  - **S:** SOLA will require SAML credentials as a security token.
  - **X:** SOLA will require an X509 certificate as a security token.
- **OutPassReqd:** with the current version of SOLA, the only option is NO.
  - **N:** SOLA will not require a password on output responses
  - **Y:** SOLA will require a password on output responses (not implemented in this release).

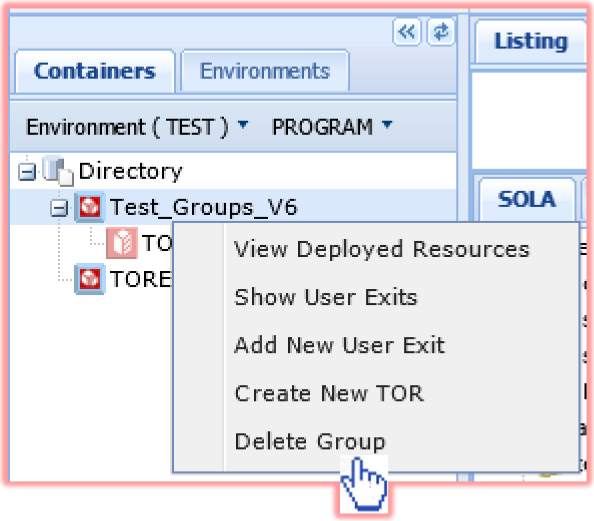
- **OutEncrType:** this setting determines how SOLA encrypts responses.
  - **N:** encryption is not required (SOLA will send responses without encryption).
  - **R3:** RSA 3DES encryption is required.
  - **R1:** RSA DES encryption is required.

- **OutSignatureReqd:** this setting determines whether SOLA will attach a signature to responses.
  - **N:** SOLA will not attach a signature to responses.
  - **Y:** SOLA will attach a signature to responses.

- **OutTimestampReqd:** this setting determines whether SOLA will attach a timestamp to responses. With the current version of SOLA, the only option is NO.
  - **N:** SOLA will not attach a timestamp to responses.
  - **Y:** SOLA will attach a timestamp to responses (not implemented in this release).

- **OutTokenReqd:** this setting determines whether SOLA will attach a token to responses. With the current version of SOLA, the only option is NO.
  - **N:** SOLA will not attach a token to responses.

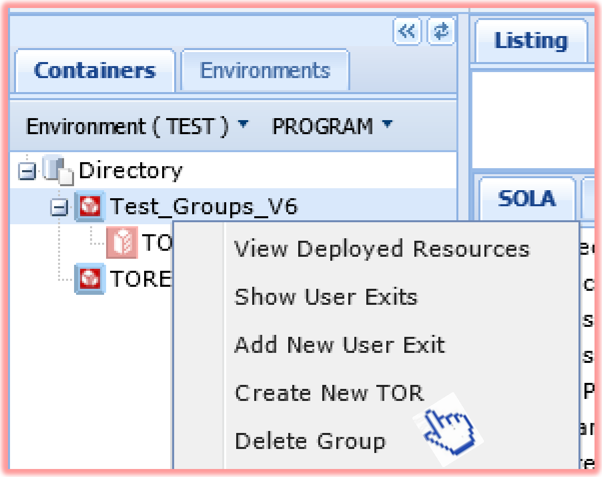
Deleting Container Groups



To delete a group, right click the group and select **Delete Group** from the menu.

Deleting a group will delete all containers in that group and all group settings (custom security policy, etc.).

Creating Containers

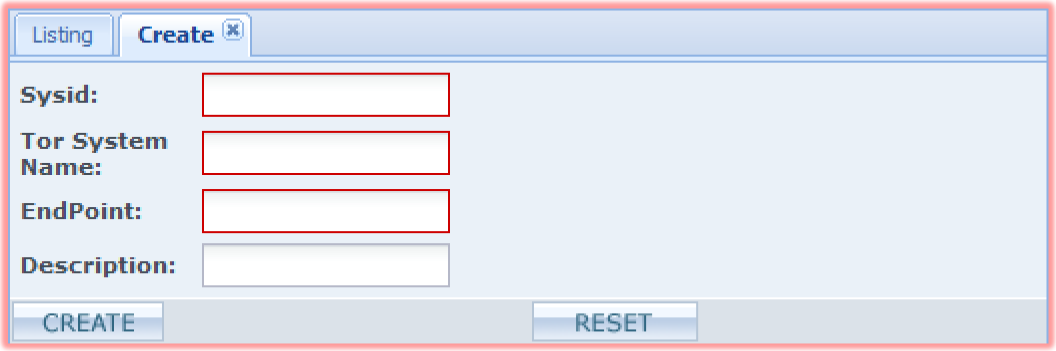


Once you have created and configured one or more container groups, you can create SOLA containers within those groups.

To create a SOLA container, right click on the group icon and select **Create New TOR** from the menu.

This will display the Create tab in the workspace, allowing you to create a new SOLA container.

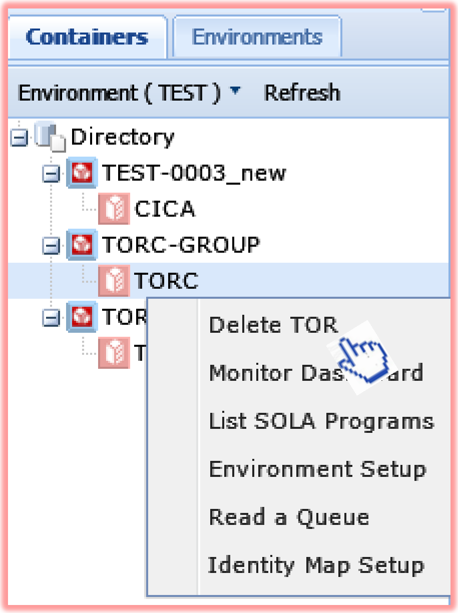
Fill out the required information about the container.



- **Sysid:** The 4 character SYSID of the CICS region
- **Tor System Name:** The 8 character Applid of the CICS region
- **EndPoint:** the region's IP address and port number that CICS Web Support is listening to. Example: HTTP://MAINFRAME.SOA.LOCAL:1453/CICS/XML/XMLPC000
- **Description**: a brief description of the region (optional).

When you have filled out all required fields, click **CREATE** to create the new container.

Deleting Containers



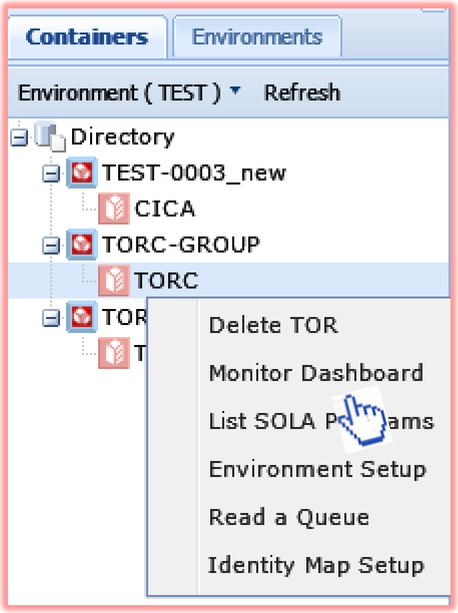
To delete a container, right click the container in the Directory tree and select **Delete TOR** from the menu.

Deleting a container will delete the container from that group and all of the containers associations including programs/methods, custom security policy, etc.).

Monitoring SOLA Containers

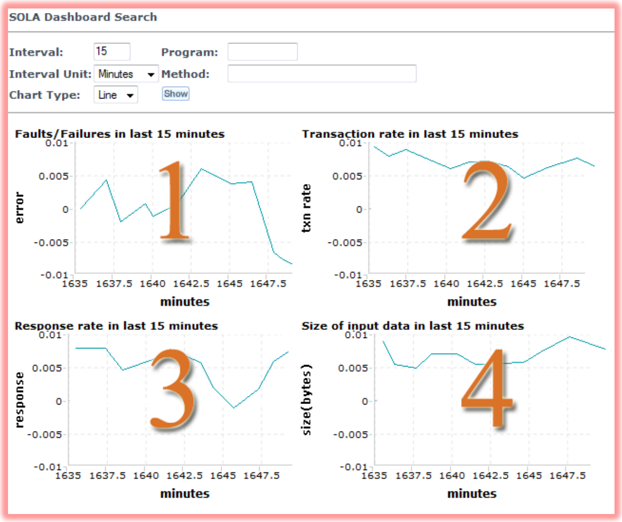
Resource Manager has an active dashboard that provides container performance metrics in real time. The metrics frequency is controlled by the dashboard, not the SOLA group settings for metrics collection.

To access the dashboard for a container, right click on the container and select **Monitor Dashboard** from the menu.



The Dashboard tab will be displayed in the workspace.

The dashboard is divided into four panels, each of which provides specific information (default settings shown).


SOLA Dashboard Search

Interval: 15
Program:
Interval Unit: Minutes
Method:
Chart Type: Line
Show

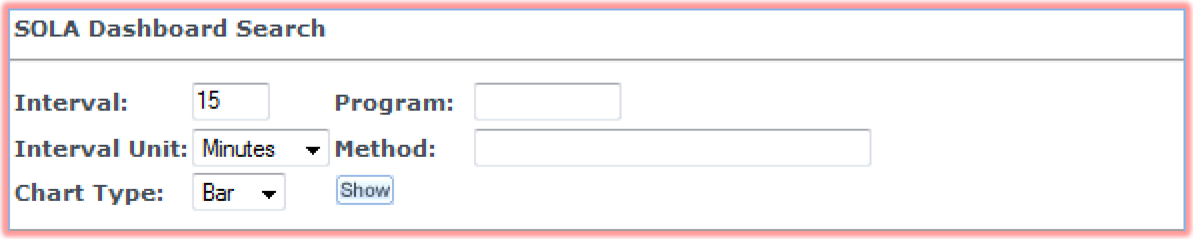
Panel 1: Faults/failure rate in the last *n minutes/seconds*.

Panel 2: Transaction rate in the last *n minutes/seconds*.

Panel 3: Response rate in the last *n minutes/seconds*.

Panel 4: Input data size over the last *n minutes/seconds*.

You can configure the dashboards using the top panel.



SOLA Dashboard Search

Interval: 15 Program:

Interval Unit: Minutes Method:

Chart Type: Bar

- **Interval:** the interval to chart, in minutes or seconds, depending on the Interval Unit menu.
- **Interval Unit:** determines whether the data collection interval is measured in minutes or seconds.
- **Chart Type:** Determines how the data is presented. Options are line or bar.
- **Program:** narrows down the data collection to a single program running in the container.
- **Method:** narrows down the data collection to a single method (operation) running in the container.

Working with Resources

Resources are used in conjunction with subjects to create accesses and active policies. Certificates are also resources, though they do not interact with subjects but are instead used on their own to control authentication.

The following resources are used in Resource Manager:

- **Programs and methods in the SOLA directory:** The SOLA directory is the same directory used by SOLA Developer. Whatever changes you make to the directory will also be seen in the SOLA Developer, and vice versa. Resource Manager uses the SOLA Directory to create accesses and assign policies by associating subjects with either programs or methods in the directory. Detailed information on how to create accesses and assign policies is discussed in the "Working with Accesses" section, which starts on page 42.

- **IPs:** IP groups are used to organize IP addresses and can make creating IP accesses easier.

- **Certificates:** there are two ways in which Resource Manager makes use of certificates:

  Trusted Certificates: In order to allow SAML tokens to be used for mainframe runtime authentication, the SAML Authority's certificate is uploaded to Resource Manager and designated as a trusted certificate, then deployed to a SOLA container group. SOLA will recognize that SAML authority and authenticate (but not authorize) accounts that use that authority's tokens.

  Untrusted Certificates: these certificates are used to authenticate application servers that consume SOLA web services. If a policy requires that all requests be signed, those signatures must have either a keyname or provide a certificate. The certificate can only be accepted if it has been uploaded to SOLA. Such certificates must be designated "untrusted".

Resource Deployment Rules

Every time a resource is deployed to a container group, Resource Manager performs a series of checks to see if the accesses or assigned policies have been updated. Those updates are then reflected in the container group.

For example, if you drag a user subject into a resource and create an access, then deploy that resource to a container group, that access will be active in that container group. If you then add a policy or additional users or any combination thereof and redeploy that same resource to the same container group, all of the additional accesses and/or policies will be activated in that container (in addition to those previously deployed). If you then remove some of the users or policies and once again deploy that resource to the same container group, the removed subjects will no longer be active in the container group. In this manner you can update your active accesses and assigned policies.

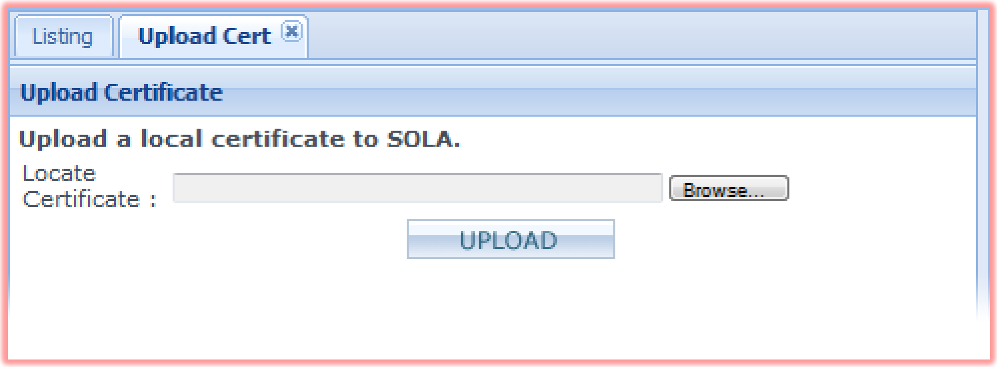
Working with Certificates

Uploading Certificates

SOLA supports a public key/private key architecture, where matching pairs of public and private keys are used. When a message is encrypted with a key, it can only be decrypted with the matching key from the key pair. If you want SOLA to encrypt your responses then you have several options:

- Include your public key on your request messages
- Upload your public key to SOLA, and tell SOLA to use that public key with the keyname token

To upload a certificate to Resource Manager, click on the root directory in the Certificates tab in the Resource panel, then select **Upload Certificate** from the menu.



Upload a local certificate to SOLA.

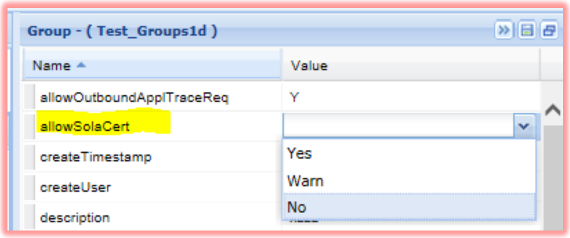
Locate Certificate :

Click **UPLOAD** when you are ready.

NOTE: if you are using Websphere 6.1, then you must convert your certificates from binary to Base64 encoded prior to uploading.

Configuring Certificates

An attribute "allowSOLACert" is captured at the container level. If the attribute is not set or not available then the default behavior is to support use of SOLA product certificate for processing encryption/decryption/signature. The attribute if captured can be set to "Yes", "Warn", "No".



Group - (Test_Groups1d)

Name	Value
allowOutboundApplTraceReq	Y
allowSolaCert	
createTimestamp	
createUser	
description	

1. Option '**Yes**' is same as current default to allow usage of SOLA Certificate.

2. Option '**Warn**' when set will allow applications to use SOLA certificate but warnings are generated in SOLA Logs. This option can help to identify applications that use SOLA default certificate.

3. Option '**No**' when set will reject the request if it uses SOLA certificate.

Downloading Certificates

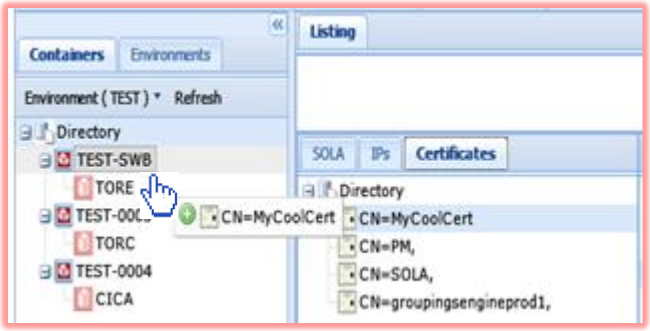
Resource Manager allows you to download the public key of any certificate that has been uploaded. To download a certificate, right click the directory root and select **Download Certificate** from the menu.

Deploying Certificates

Once a certificate has been uploaded and configured, it is not active until it is deployed. The deployment of a certificate is accomplished using Resource Manager's drag and drop capabilities; a certificate is dragged over to a SOLA container, activating that certificate for every container in the target container's group.

NOTE: although Resource Manager does not allow subjects to be dragged into container groups, a subject (group) can only be activated for a container group, not individual containers. Dragging a certificate into a container group deploys that certificate to every container in the container group.

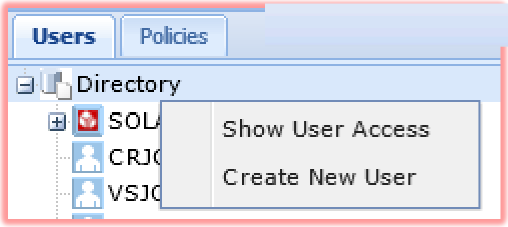
The following illustration shows a certificate being deployed in the TEST-SWB container group.



Working with Accesses

An access is a subject (such as a user or policy) that is associated with a resource (such as a program, method, IP address or certificate). Once associated, the subject has “invoke” access to the resource. Resource Manager is only concerned with run-time access, so “invoke” access (ability to run a program) is the only access it grants.

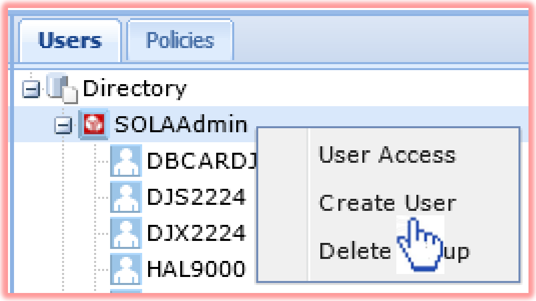
Creating and Managing Users



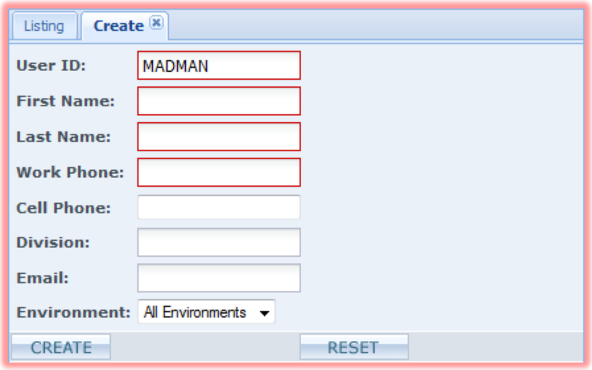
To create a new user, right click the directory root and select **Create New User** from the menu.

This will display the Create tab and allow you to create a new user account by filling in the user account name and contact information for the user. Fields outlined in red are required.

To create a user account in a specific group (such as SOLAAdmin), right click the group name instead of the directory root and select **Create User** from the group menu.



Creating a new user in Resource Manager is identical to creating a user account in SOLA Developer, with one minor difference; the access level of the account (either programmer or administrator) depends on whether or not the user is in the **SOLAAdmin** group (see below).

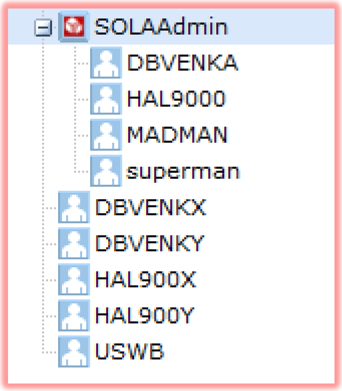


User ID: MADMAN
First Name:
Last Name:
Work Phone:
Cell Phone:
Division:
Email:
Environment: All Environments

User accounts created in Resource Manager will be available for use in SOLA Developer, and users created in SOLA Developer will likewise be available in Resource Manager and will appear in the users tab in the subjects panel.

Resource Manager will allow you to create users that exist only in specific environments by selecting **Environment** from the Create tab, or selecting “All Environments” the User account will be made available in all environments.

User Groups



- SOLAAdmin
  - DBVENKA
  - HAL9000
  - MADMAN
  - superman
- DBVENKX
- DBVENKY
- HAL900X
- HAL900Y
- USWB

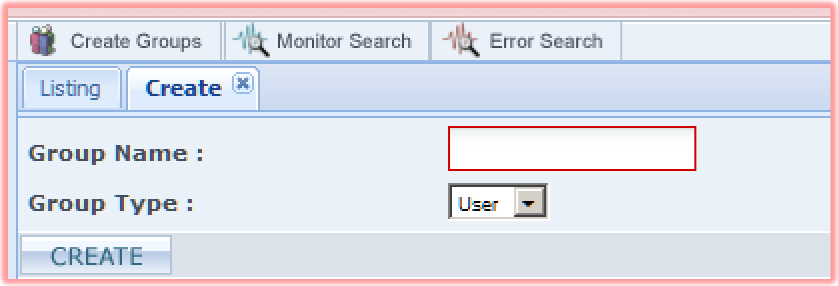
User groups serve two purposes; organizing users to create multi-user accesses and determining which users have administrator access.

For the latter purpose, Resource Manager comes with a default group called "SOLAAdmin". Users created in that group are SOLA Administrators with full administrative access rights to SOLA Developer.

If you delete the SOLAAdmin group, you will delete all users in that group and delete all SOLA Administrator accounts. If the group has been deleted, you can create a new group called "SOLAAdmin", and that group will function exactly as the original system group.

Other than the SOLAAdmin group, groups serve no purpose other than to organize user accounts and make creating user accesses easier.

You can drag a user from the root directory or another group into a new user group, as well as drag a user out of a group and into the root directory.



Group Name :

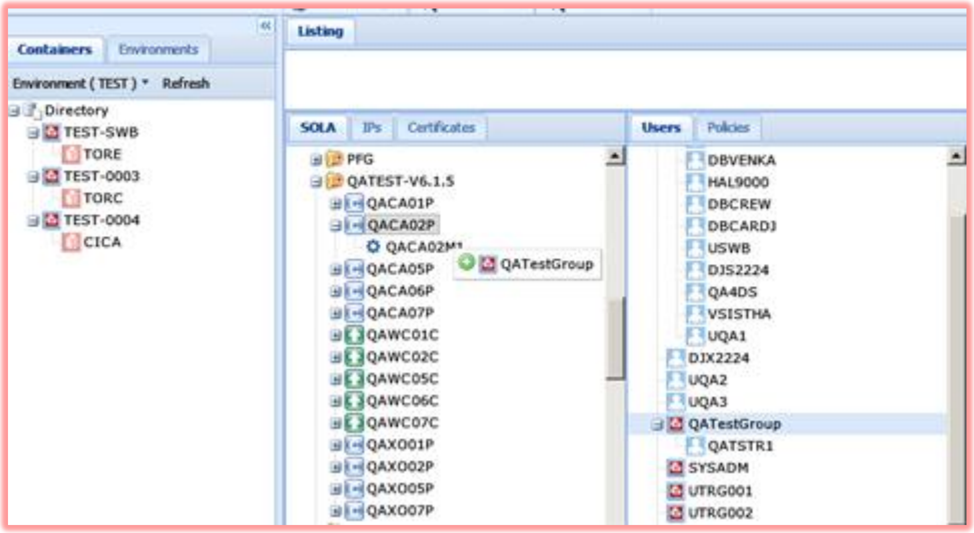
Group Type : User

To create a new user group, click the Create Groups button. This will display the Create tab and allow you to create a group.

Select "User" from the **Group Type** menu, then enter the group name. When you are ready to create the group, click the CREATE button.

Creating a User Access

Once you have populated the Users tab and determined which Users will need Policy Admin and/or Tester Admin rights, you can start to create accesses. An access is created when a User group is associated with a resource. This association is accomplished using Resource Manager's drag and drop capabilities; a User group is dragged over to a resource and dropped into it.

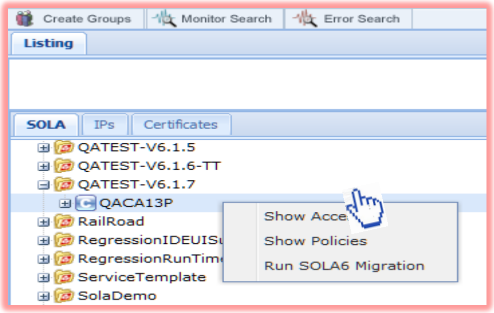


In the following illustration, Users group **QATestGroup** is being dragged to program QACA02P. The resulting access, once it is deployed, will allow anyone in User group **QATestGroup** to invoke that program and/or method.

The following table illustrates the effects of various associations:

Subject	Resource	Result
User Group	Program	All users can invoke any method in the target program.
User Group	Method	All users can invoke the target method.

Once an access is created, you can view it by right clicking on a resource and selecting **Show Accesses**.



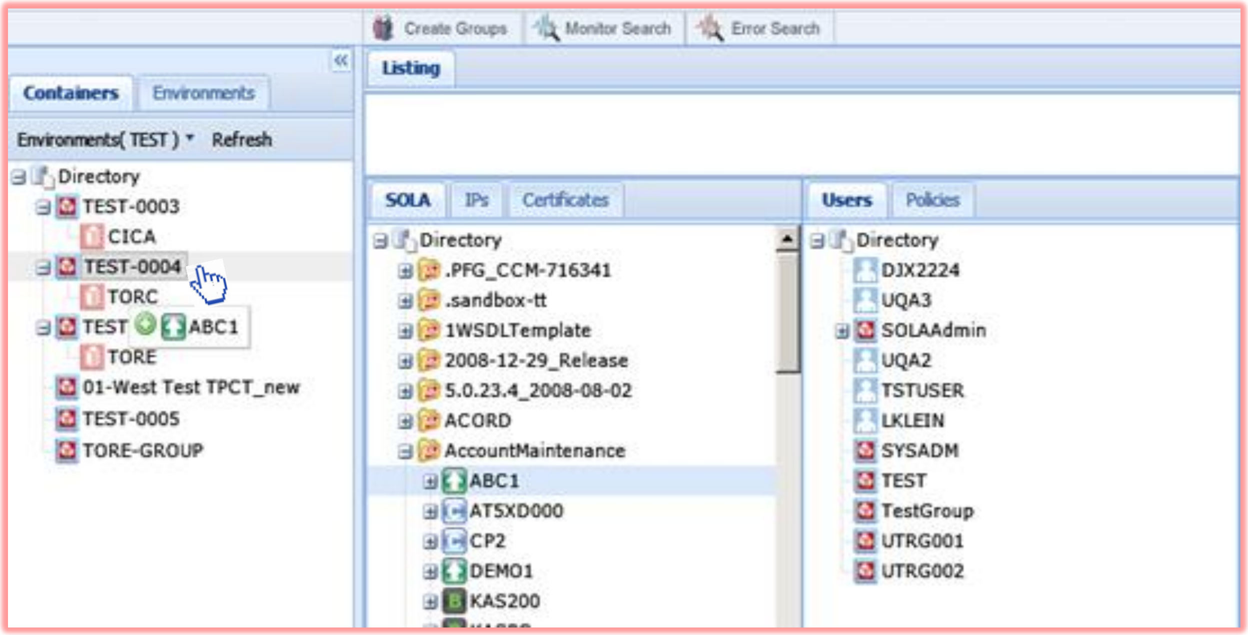
Deploying a User Access

When a User access is created, it is not active until it is deployed. For example, if you drag a user from the subjects panel to a program in the resource panel, the access you create is not in effect; that user does not have invoke rights to the program. To activate the access, you must deploy that access into a SOLA container group.

The activation of an access is accomplished using Resource Manager's drag and drop capabilities; an access is dragged to a SOLA container group; deploying allows access for every container in the target container's group.

NOTE: although Resource Manager does allow accesses to be dragged into container groups, an access can only be activated for a container group, not individual containers.

The following illustration shows program **ABC1** being deployed in the **TEST-0004** container group.



Creating and Managing IP Addresses

For the purpose of this IP Address facility, the following terms are defined as follows:

Container: A container is an environment in which the SOLA runtime is deployed. When working with CICS, the SOLA runtime container runs in a TOR (Terminal Owning Region). Resource Manager gives you the ability to use an existing TOR region as a SOLA container by registering that region to SOLA.

Master: Represents the container for SOLA Developer & Resource Manager.

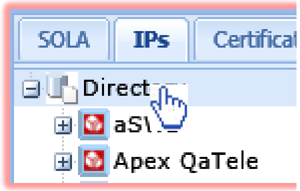
Runtime: Represents the actual containers defined within a particular group and acts as a gateway to your application programs.

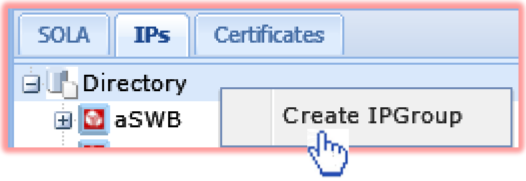
IP Groups

IP groups are used to organize IP addresses and represent the object the user will actually deploy to a particular runtime environment. Instead of dragging / dropping individual IP addresses onto a deployable resource, you will now drag / drop the entire IP Group onto the deployable resource (*container group*). This has the effect of creating access for all the IP addresses in that group within the particular runtime environment.

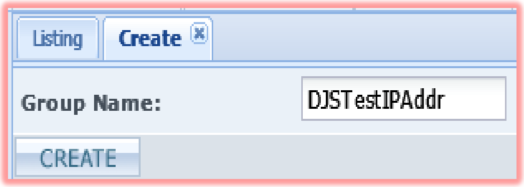
When creating an IP group, the IP addresses in that group, once deployed to a particular runtime, will have full access to all SOLA programs in all SOLA containers defined within that container group. This IP facility allows container level access (accessAllInd = 'Yes') to the WOR region, and currently does not offer more granular restrictions.

To create a new IP Universal Access group, right click on the Directory in the tree and select **Create IPGroup.**



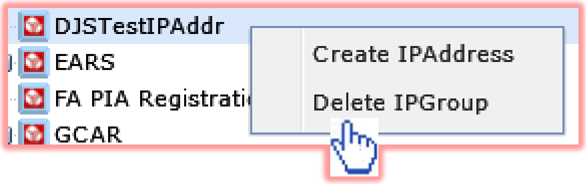


Fill in the Group Name and click the **Create** button. A dialog box will indicate the IP Group was created successfully; click **OK.**



Group Name: DJSTestIPAddr

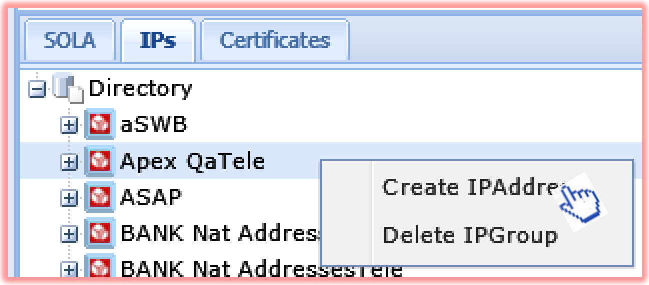
To delete an IP Group, right click the IP group name and select **Delete IPGroup.**



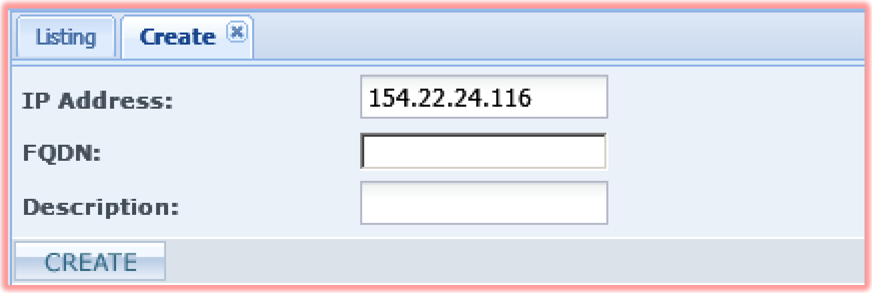
A dialog box will indicate the IP Group was deleted successfully; click **OK.**

Creating an IP Address

To create an IP Address in a specific group, right click the group name and select **Create IPAddress** from the group menu.



To create an IP address, fill in the following (only the **IP Address** field is required):



IP Address: 154.22.24.116

FQDN:

Description:

- **IP Address:** the IP address (xxx.xxx.xxx.xxx) that you want to authorize. This will typically be the IP address of a client machine that wants to access SOLA web services. Please note that trailing nodes within the IP Address may be entered as wildcards with the asterisk(*). For example: **10.5.20.*** or **10.5.20.3*** or **10.5.21*.***

- **FQDN:** the fully qualified domain name (used only for documentation purposes).

- **Description:** an optional description of the IP address (e.g. "OrdersApplicationServer").

When you have completed setting up the IP address data click the CREATE button to create the IP address. A dialog box will indicate the IPAddress was created successfully; click **OK.**

Deleting an IP Address

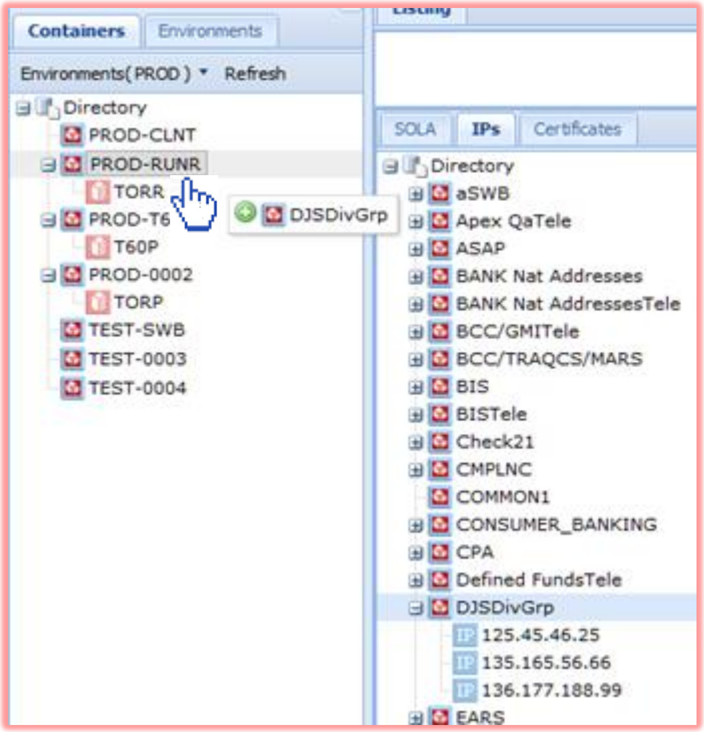
To delete an IP Address in a specific group within the Runtime database, right click the group name and select **Delete IPGroup.** A dialog box will indicate the IPAddress was deleted successfully; click **OK.**

Deploying an IP Address

Deploying to a runtime environment entails dragging / dropping an IP Group to a particular Container Group within the container panel on the left.

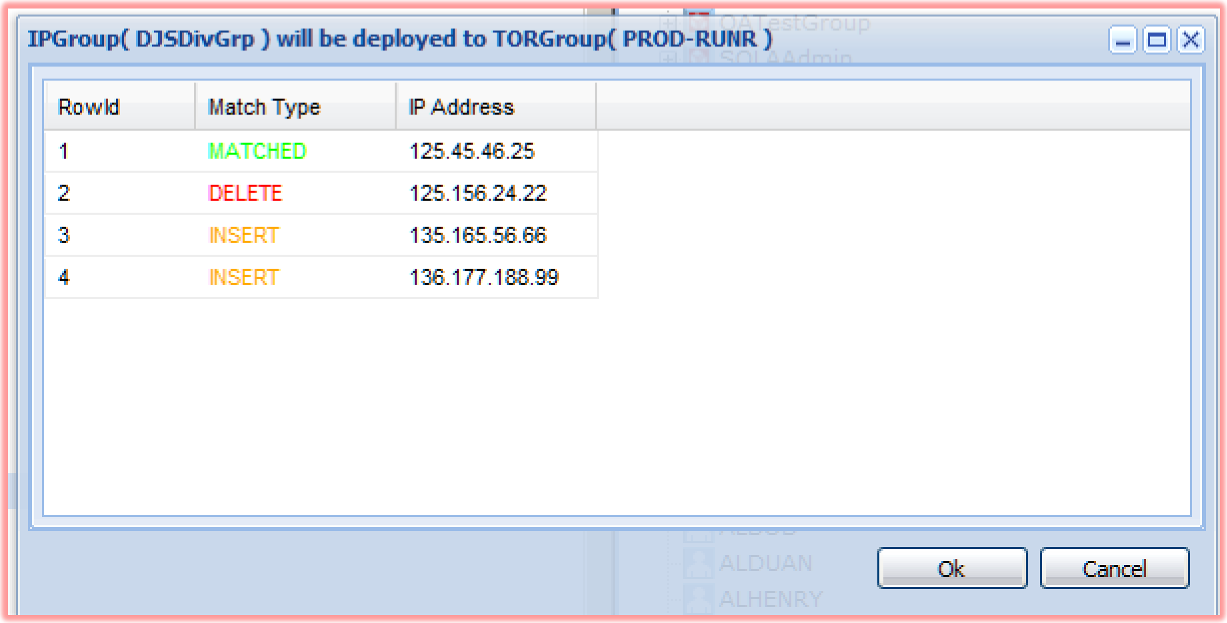
In the following illustration IP Group '**DJSDivGrp**' is being dragged / dropped to Container group '**PROD-RUNR**'.

Once an IP Group is dragged / dropped and deployed to the Container group a dialog box will appear with a reconciliation list detailing the actions that will be taken in order to synchronize the runtime container with the IP Addresses you have defined within the master. This will provide you with an opportunity to review the updates being made, before actually committing the changes.



If the list exceeds the available screen size, then you will need to scroll to see all of the IP Addresses in the IP Group and their associated maintenance types.

The information is organized under a series of column headings:


IPGroup(DJSDivGrp) will be deployed to TORGroup(PROD-RUNR)

RowId	Match Type	IP Address
1	MATCHED	125.45.46.25
2	DELETE	125.156.24.22
3	INSERT	135.165.56.66
4	INSERT	136.177.188.99

The following labels that appear in the confirmation report are interpreted as follows:

- **Match Type:**

  MATCHED = IP Address exists in both the Master and Runtime databases and will remain unchanged (no action for this IP address will be taken).

  DELETE = IP Address exists in the Runtime, but not in the Master and will be deleted from the Runtime.

  INSERT = IP Address exists in the Master, but not in the Runtime and will be added to the Runtime.

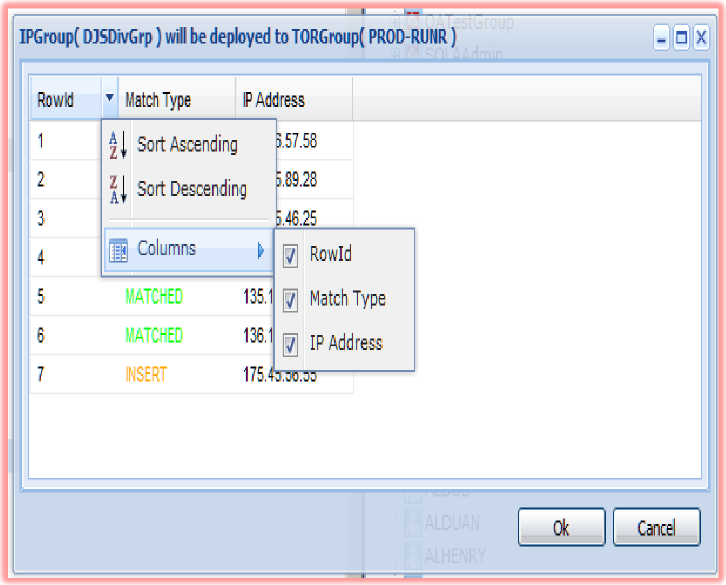
- **IP Address:** the IP Address number to be authorized for access

In the above illustration:

- RowID 1 was an existing IP Address and will remain untouched
- The IP Address identified in RowID 2 is being deleted
- RowID 3 and 4 were the new IP Addresses just added

After verifying the changes click **OK** to finalize deployment of the Resource IPGroup(DJSDivGrp) to the Container TORGroup(PROD-RUNR). This will synchronize the Master database with the Runtime database. If you choose to click Cancel no updates to the Master or the Runtime will take place.

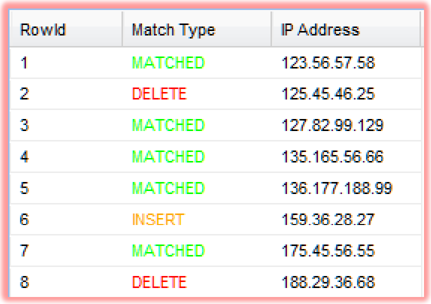
You can change the order of the data in the list and hide columns by selecting the drop down arrow in either of the column headings:



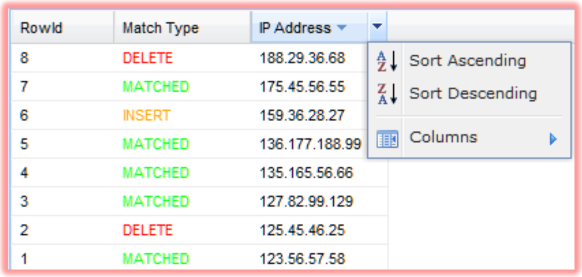
IPGroup(DJSDivGrp) will be deployed to TORGroup(PROD-RUNR)

RowId	Match Type	IP Address
1		5.57.58
2		5.89.28
3		5.46.25
4		
5	MATCHED	135.1
6	MATCHED	136.1
7	INSERT	175.4[illegible]

This view is presented immediately following deployment. After changing the Sort order to Sort Descending you will notice the IP Address order is reversed:

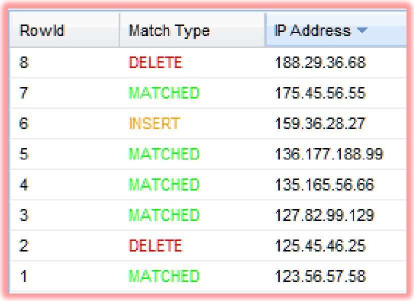


RowId	Match Type	IP Address
1	MATCHED	123.56.57.58
2	DELETE	125.45.46.25
3	MATCHED	127.82.99.129
4	MATCHED	135.165.56.66
5	MATCHED	136.177.188.99
6	INSERT	159.36.28.27
7	MATCHED	175.45.56.55
8	DELETE	188.29.36.68

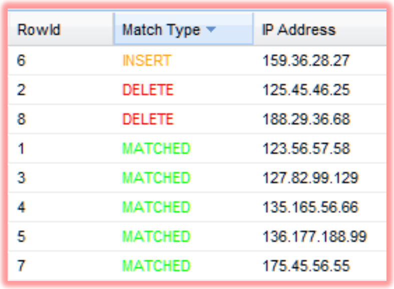


RowId	Match Type	IP Address
8	DELETE	188.29.36.68
7	MATCHED	175.45.56.55
6	INSERT	159.36.28.27
5	MATCHED	136.177.188.99
4	MATCHED	135.165.56.66
3	MATCHED	127.82.99.129
2	DELETE	125.45.46.25
1	MATCHED	123.56.57.58

The view on the left below is presented immediately following deployment. After changing the Match Type order to Sort Descending the view located on the right will list all **INSERT**/**DELETE** items first followed by all **MATCHED** IP Addresses that will remain unchanged.



RowId	Match Type	IP Address
8	DELETE	188.29.36.68
7	MATCHED	175.45.56.55
6	INSERT	159.36.28.27
5	MATCHED	136.177.188.99
4	MATCHED	135.165.56.66
3	MATCHED	127.82.99.129
2	DELETE	125.45.46.25
1	MATCHED	123.56.57.58



RowId	Match Type	IP Address
6	INSERT	159.36.28.27
2	DELETE	125.45.46.25
8	DELETE	188.29.36.68
1	MATCHED	123.56.57.58
3	MATCHED	127.82.99.129
4	MATCHED	135.165.56.66
5	MATCHED	136.177.188.99
7	MATCHED	175.45.56.55

Working with Policies

Resource Manager gives you two ways to create policies and apply them to SOLA containers, programs and methods. The first way is a default policy, and it was covered in the section on creating container groups (page 25). When creating a group, you can create a default policy that will apply to every program running in every container in that group.

In addition to a default policy, you can create a policy in the subjects panel, then apply that policy to any resource or group of resources (except certificates), thereby creating an active policy. You can then deploy that active policy to a container group. It is important to understand that all resources and subjects are administered on the Runtime database. This database is referred to in the Developer User Guide as the **Program(MSTR)**. The active policy will override the default policy, but only for the resources that it is applied to. Also, the default policy may have additional detail that will not be overridden.

For example, the policy you create in the subjects' panel may specify that a username token is required, but does not specify what kind. The default policy, on the other hand, requires a SAML token. In such an instance, the default policy will be used to determine the type of token that is required.

A good rule of thumb is that if the default policy and
the dominant subject panel policy do not conflict, and the default policy offers a specific definition of what is required by the subject panel policy, the default policy's definition will be enforced.

Creating and Managing Policies

Creating policies works a bit differently than creating other subjects such as users and IP addresses. For all other subjects, the individual tree item is the subject, and the group is a container for the subjects. With policies, the group is the policy, and the items in a group are aspects of that policy.

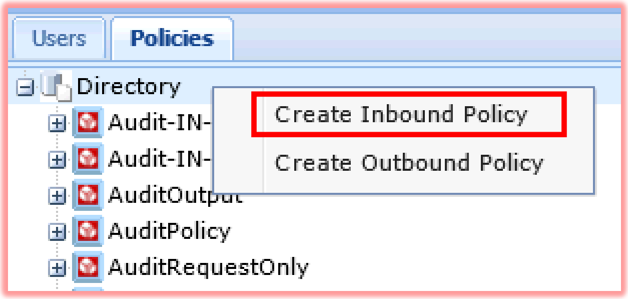
Within the policy tree itself, the only allowable drag and dropping of policies is to associate aspects of a policy with a group, or assign them to a resource.

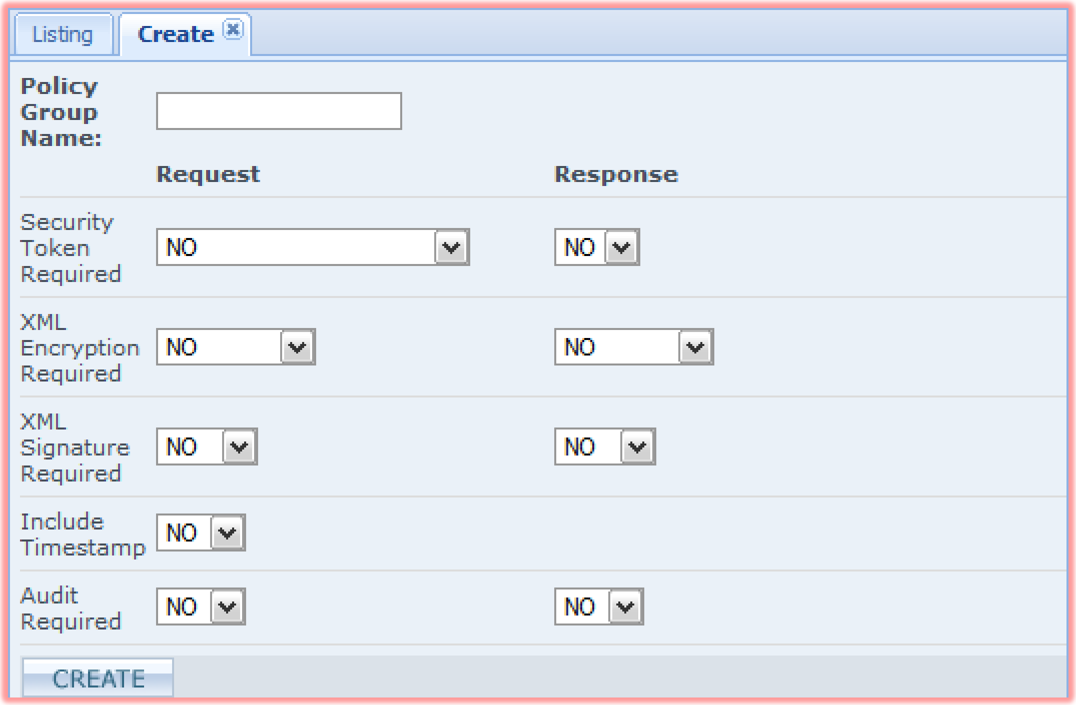
SOLA supports 2 types of policies

- Inbound Policy to be applied on all Inbound programs
- Outbound Policy to be applied on all Outbound programs

Creating and Managing Inbound Policies

To create a new inbound policy, right click the directory root and select **Create Inbound Policy** from the menu. This will display the Create tab, allowing you to create a new policy.





	Request	Response
Policy Group Name:		
Security Token Required	NO	NO
XML Encryption Required	NO	NO
XML Signature Required	NO	NO
Include Timestamp	NO	
Audit Required	NO	NO

Enter a policy name in the Policy Group Name field, then configure the policy using the options shown below.

Input (Request) Settings

Security Token Required: this setting determines whether SOLA will accept requests without an attached security token. The type of token required can be defined in a container group policy (see page 25).

- **NO:** SOLA will allow requests without security tokens
- **Username Token:** SOLA will only accept requests with username tokens.
- **Encrypted Username Token:** SOLA will only accept requests with encrypted username tokens.
- **SAML Token:** SOLA will require SAML credentials as a security token.

XML Encryption Required: this setting determines whether SOLA accepts requests that are not encrypted.

- **NO:** encryption is not required (SOLA will accept requests without encryption).
- **RSA-3DES:** encryption is required, and must be RSA-3DES (more schema options will be available with future versions of SOLA).

XML Signature Required: this setting determines whether SOLA accepts requests without an attached signature.

- **NO:** attached signatures are not required (SOLA will accept requests without attached signatures).
- **Body:** the body of the SOAP request must be signed.

Include Timestamp: this setting determines whether SOLA accepts requests without an attached timestamp. The timestamp contains the policy's expiration date and time.

- **NO:** attached timestamps are not required (SOLA will accept requests without attached timestamps).
- **YES:** attached timestamps are required.

Audit Required: this setting determines whether SOLA will trace input XML, thereby auditing request.

- **NO:** instructs SOLA not to audit input XML.
- **YES:** instructs SOLA to trace the input XML, thereby auditing requests.

Output (Response) Settings

Security Token Required: this setting determines whether SOLA will attach a security token to responses. With the current version of SOLA, the only option is NO.

XML Encryption Required: this setting determines whether SOLA will encrypt responses.

- **NO:** encryption is not required (SOLA will not encrypt responses).
- **RSA-3DES:** encryption is required, and SOLA will use RSA-3DES (more schema options will be available with future versions of SOLA).

XML Signature Required: this setting determines whether SOLA will attach a signature to responses.

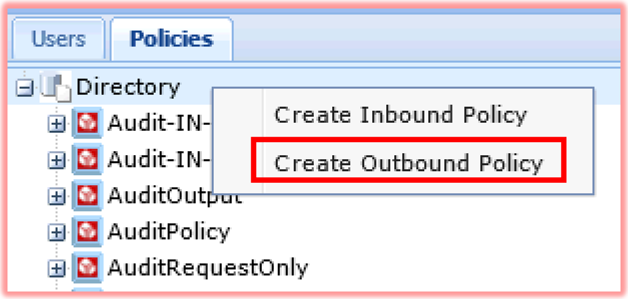
- **NO:** SOLA will not attach signatures to responses.
- **Body:** SOLA will attach a signature to the body of responses.

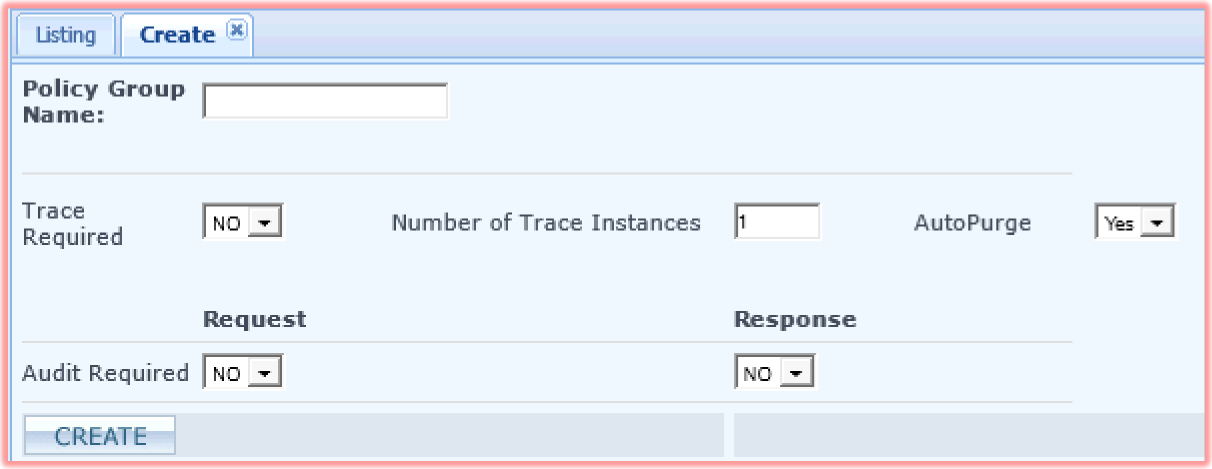
Audit Required: this setting determines whether SOLA will trace output XML, thereby auditing responses.

- **NO:** instructs SOLA not to trace output XML.
- **YES:** instructs SOLA to trace output XML, thereby auditing responses.

Creating and Managing Outbound Policies

To create a new outbound policy, right click the directory root and select **Create Outbound Policy** from the menu. This will display the Create tab, allowing you to create a new policy.





Policy Group Name:

Trace Required NO

Number of Trace Instances 1

AutoPurge Yes

	Request	Response
Audit Required	NO	NO

Enter a policy name in the Policy Group Name field, then configure the policy using the options shown below.

Trace Required: this setting determines whether SOLA Outbound Tracing is Required or not.

- **NO:** SOLA will not generate outbound trace
- **Yes:** SOLA will generate outbound trace. In CICS, SOLA trace will be generated in a TSQ having naming convention
  ST-<ProgramName>-nnnn
  Where <ProgramName> is the directory name of the Outbound program captured in SOLA and nnnn is the running sequence number of the trace

Number of Trace Instances : this setting indicates the number of trace instances to be captured. Maximum number of trace instances that can be setup is 9999.

AutoPurge: this setting determines whether trace entries captured by SOLA needs to be auto purged

- **NO:** SOLA will not Autopurge trace entries. If tracing is already captured for maximum number of trace instances as setup in the policy then further tracing will be disabled
- **Yes:** SOLA will Autopurge trace entries. If tracing is already captured for maximum number of trace instances as setup in the policy then oldest trace instance is purged and new trace instance is generated.

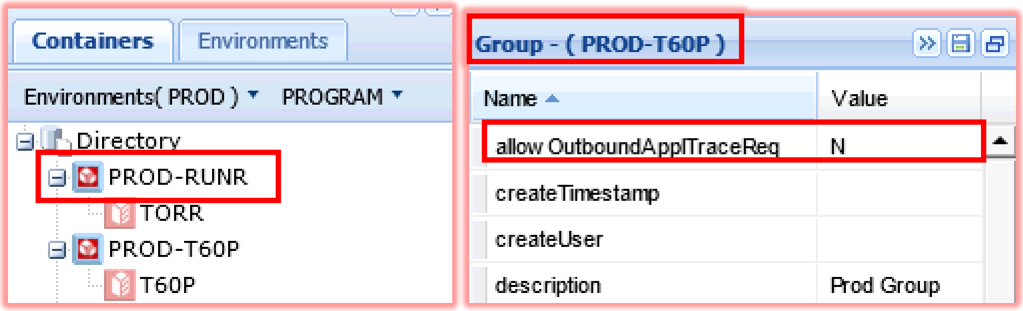
Audit Required: this setting determines whether SOLA will audit SOAP request and/or response XML.

- **NO:** instructs SOLA not to audit
- **YES:** instructs SOLA to audit

Note: *Outbound Policy Support is currently only available for applications that invoke SOLA outbound plugin under CICS.*

Managing Application enabled SOLA Outbound Tracing

SOLA outbound plugin supports application enabled tracing where by application sets WSC-INVOKE-TRACE field in the generated interface copybook area to ‘Y’. This enables tracing and trace under CICS is generated under SOLATRACE TSQ. To disable application enabled tracing on a container, select the container group and update the property “allowOutboundApplTraceReq” to “N” and save the changes as shown in the following illustration. This action will disable any application enabled tracing. Tracing is only enabled with the assignment of outbound tracing policy.



Name	Value
allowOutboundApplTraceReq	N
createTimestamp	
createUser	
description	Prod Group

Assigning a Policy

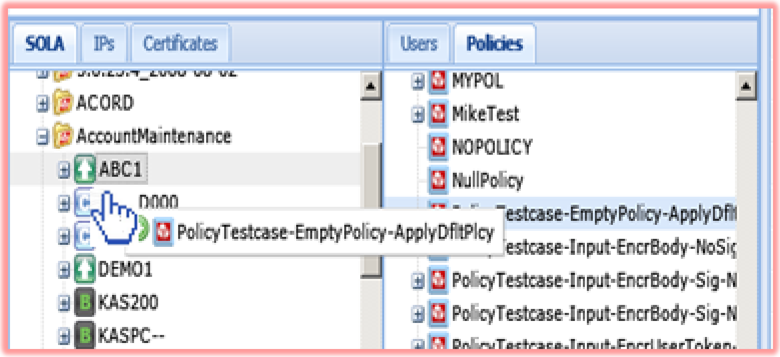
TECH TIP

SOLA™

SOLA Developer Studio has a new Policy Management function enabled as a sub-menu under the context of a Program. This interface is a more intuitive interface for managing assignment/deployment of policies

Once you have a policy, you can assign the policy to a resource or group of resources in the resource panel. This assignment is accomplished using Resource Manager’s drag and drop capabilities; a subject is dragged over to a resource and dropped into it.

In the following illustration, a policy is being dragged to program ABC1. The policy will be assigned to the program and all of its methods.



The following table illustrates the effects of various associations:

Subject	Resource	Result
Policy	Program	The policy will be assigned to the program and all of its methods. Once deployed, the program will use this policy, overriding the container default policy, except where the default policy defines a requirement set by the assigned policy.

Policy	Method	The policy will be assigned to the method only. Once deployed, the method will use this policy, overriding the container default policy, except where the default policy defines a requirement set by the assigned policy. If a policy is applied at both Program and Method level then Method level policy overrides the Program level policy

Once a policy is assigned to a resource, you can view it by right clicking on the resource and selecting **Show Policies**.

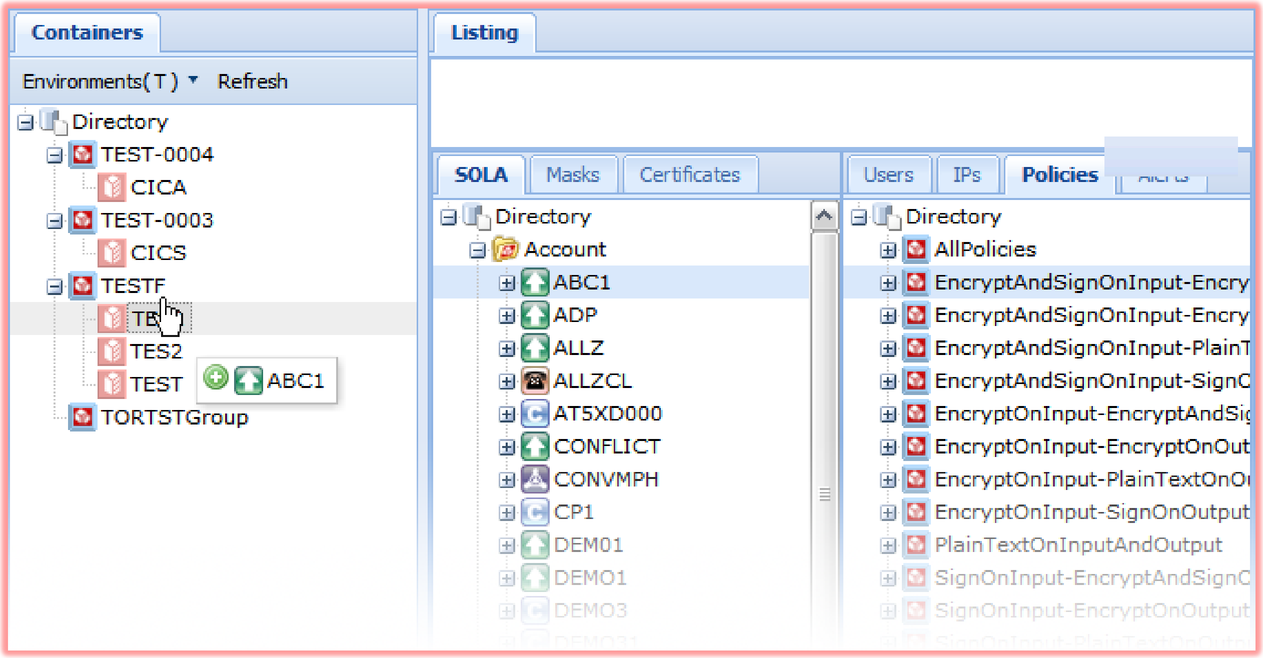
Deploying a Policy

When a policy is assigned to a resource, it is not active on a runtime container until it is deployed. For example, if you drag a policy from the subjects panel to a program in the resource panel, the policy is assigned to that resource, but is not in effect. To put it into effect, you must deploy that assigned policy into a target SOLA runtime container group.

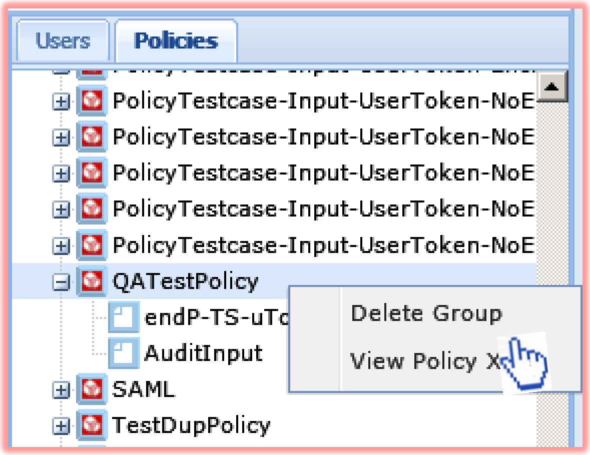
The deployment of an assigned policy is accomplished using Resource Manager's drag and drop capabilities; an assigned policy is dragged over to a SOLA container, activating that assigned policy for every container in the target container's group.

NOTE: although Resource Manager does not allow subjects to be dragged into container groups, a subject can only be activated for a container group, not individual containers. Dragging an assigned policy into a container deploys that assigned policy to every container in that container's parent group.

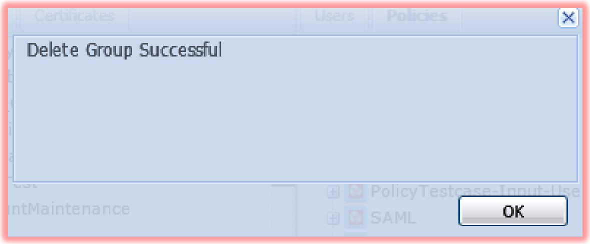
The following illustration shows a policy being deployed in the TESTF container group. It could have been dragged to any container in the TESTF group with the same results.


Containers
Environments(T)
Refresh
Directory
TEST-0004
CICA
TEST-0003
CICS
TESTF
TE
TES2
TEST
ABC1
TORTSTGroup
Listing
SOLA
Masks
Certificates
Directory
Account
ABC1
ADP
ALLZ
ALLZCL
AT5XD000
CONFLICT
CONVMPH
CP1
DEM01
DEMO1
DEMO3
Users
IPs
Policies
Directory
AllPolicies
EncryptAndSignOnInput-Encry
EncryptAndSignOnInput-Encry
EncryptAndSignOnInput-PlainT
EncryptAndSignOnInput-SignO
EncryptOnInput-EncryptAndSi
EncryptOnInput-EncryptOnOut
EncryptOnInput-PlainTextOnO
EncryptOnInput-SignOnOutput
PlainTextOnInputAndOutput
SignOnInput-EncryptAndSignO
SignOnInput-EncryptOnOutput

Deleting a Policy Group


Users
Policies
PolicyTestcase-Input-UserToken-NoE
PolicyTestcase-Input-UserToken-NoE
PolicyTestcase-Input-UserToken-NoE
PolicyTestcase-Input-UserToken-NoE
PolicyTestcase-Input-UserToken-NoE
QATestPolicy
endP-TS-uTo
AuditInput
SAML
TestDupPolicy
Delete Group
View Policy X

To delete a group, right click the group and select **Delete Group** from the menu.

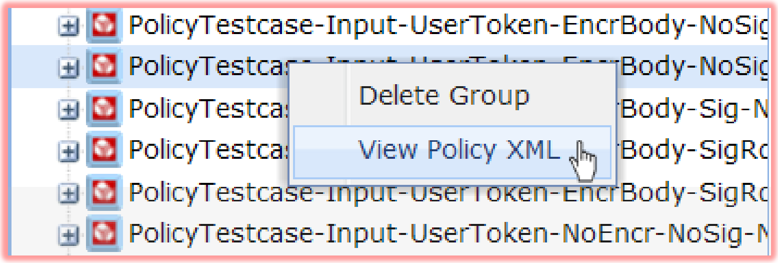


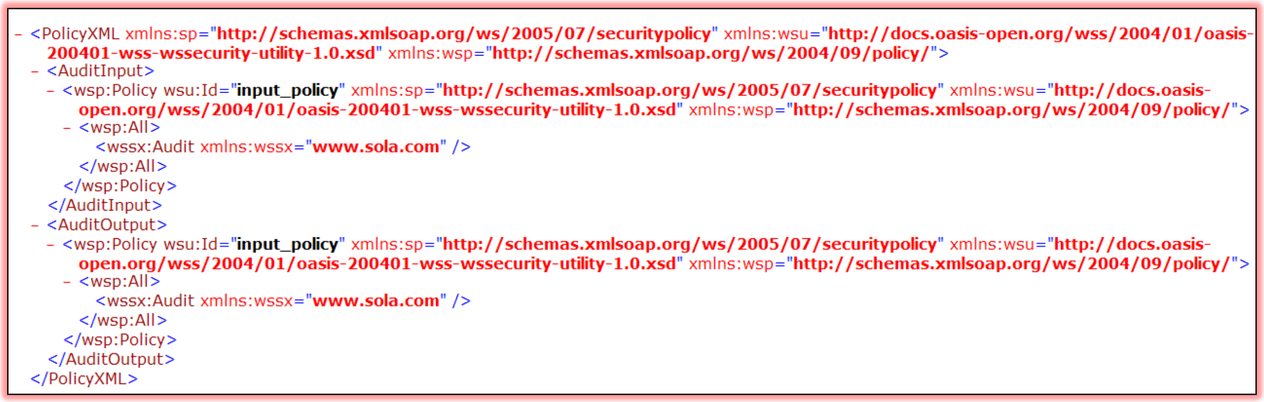
Delete Group Successful

Deleting the Policy will also delete all associated Policy aspects.

Viewing Policy XML

You can view a Policy assertion by right clicking on a Policy to bring up the Policy pop-up menu. Clicking on View Policy XML will display the XML Policy assertion in a separate browser window.

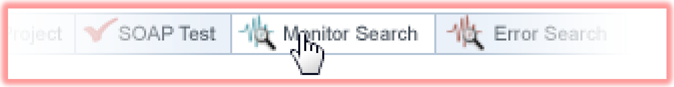

PolicyTestcase-Input-UserToken-EncrBody-NoSig
PolicyTestcas
Body-NoSig
Delete Group
View Policy XML
PolicyTestcas
Body-Sig-N
PolicyTestcas
Body-SigRc
PolicyTestcase-Input-UserToken-EncrBody-SigRc
PolicyTestcase-Input-UserToken-NoEncr-NoSig-N



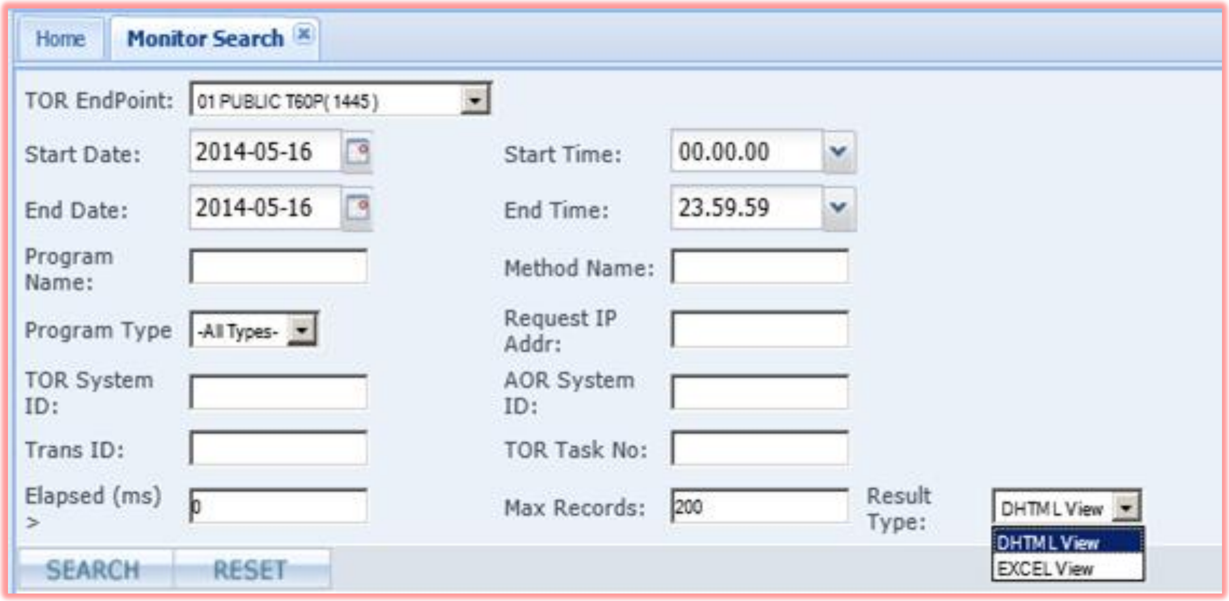
```
- <PolicyXML xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-
   200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy/">
 - <AuditInput>
   - <wsp:Policy wsu:Id="input_policy" xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy" xmlns:wsu="http://docs.oasis-
       open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy/">
     - <wsp:All>
         <wssx:Audit xmlns:wssx="www.sola.com" />
       </wsp:All>
     </wsp:Policy>
   </AuditInput>
 - <AuditOutput>
   - <wsp:Policy wsu:Id="input_policy" xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy" xmlns:wsu="http://docs.oasis-
       open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy/">
     - <wsp:All>
         <wssx:Audit xmlns:wssx="www.sola.com" />
       </wsp:All>
     </wsp:Policy>
   </AuditOutput>
 </PolicyXML>
```

Transaction Logs

You can search the transaction logs by clicking the **Monitor Search** button on the button bar.



This will display the monitor search panel.



TOR EndPoint:	01 PUBLIC T60P(1445)				
Start Date:	2014-05-16	Start Time:	00.00.00		
End Date:	2014-05-16	End Time:	23.59.59		
Program Name:		Method Name:			
Program Type	-All Types-	Request IP Addr:			
TOR System ID:		AOR System ID:			
Trans ID:		TOR Task No:			
Elapsed (ms) >	0	Max Records:	200	Result Type:	DHTML View

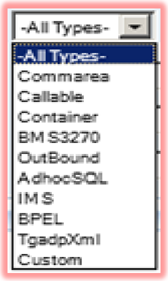
To conduct a search of the transaction log, enter search parameters using the search fields to narrow the scope of your search. You can also conduct a search with the default (mostly blank) settings, though this may take some time to complete and may result in a very long list of transactions.

The following is a description of the search fields:

- **TOR EndPoint:** narrows the search to transactions within a matching TOR region.
- **Start Date and End Date:** the start and end dates are automatically populated with the current date, though these values can be changed if necessary. All transactions are stamped with the date and time at which they take place and only transactions that took place on or after the start date and on or before the end date will be returned.
- **Start Time and End Time:** the start and end times are automatically populated with the current system time and can be changed by manually entering a time (hh.mm.ss). All transactions are stamped with the date and time at which they take place, and only transactions that took place at or after the start time and at or before the end time will be returned.
- **Program Name:** narrows the search to transactions executing this program.

- **Method Name:** narrows the search to transactions generated by the execution of the specified method.

- **Program Type:** narrows the search to transactions initiated by a method executed by the specified program type. Options are **All Types** listed in the illustration to the right:



- **Request IP Addr:** narrows the search to transactions generated in response to a request that originated from an IP address matches the specified IP address (if the request came via HTTP).

- **TOR System ID:** narrows the search to transactions with a matching TOR system Id.

- **AOR System ID:** narrows the search to transactions with a matching AOR system Id.

- **Trans ID:** narrows the search to transactions with a matching transaction Id.

- **TOR Task No:** unique identifier that is given to each unique instance of a program running in a TOR.

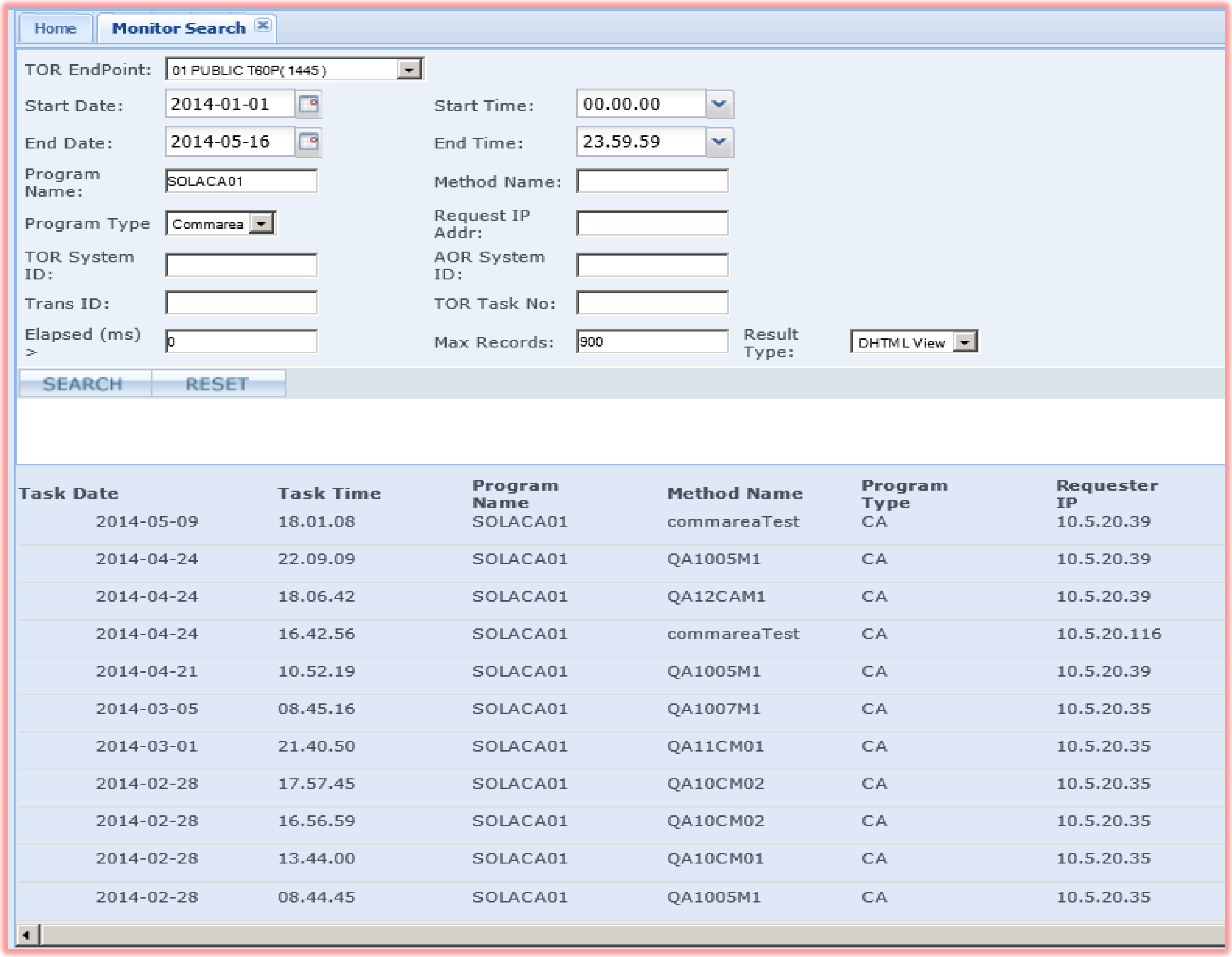
- **Elapsed (ms):**   filter's the search to transaction's that are long running based on 'Elapsed (ms)' by specifying the threshold for filtering the records based on task elapsed time.

- **Max Records:** specify Max number of records (up to 9999) that need to be extracted for analyzing monitoring data. Monitoring data can be extracted into an Excel spreadsheet with **Result Type** 'EXCEL view'. This gives the flexibility to the administrator to exploit Excel based filtering, pivoting & graphing tools to mine into the monitoring statistics.

- **Result Type:** specifies how the results will be displayed, either as DHTML (normal view) or as an Excel spreadsheet. Selecting Excel will download the results and open MS Excel (if installed), displaying the data in an Excel spreadsheet.

Once you have specified your search parameters, click SEARCH.

The results of the search will be displayed below the monitor search panel. If the list exceeds the available screen size, then you will need to scroll to see all of the search results.

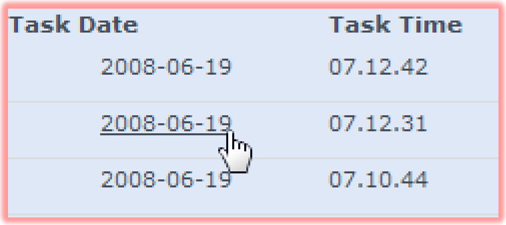


TOR EndPoint:	01 PUBLIC T60P(1445)				
Start Date:	2014-01-01	Start Time:	00.00.00		
End Date:	2014-05-16	End Time:	23.59.59		
Program Name:	SOLACA01	Method Name:			
Program Type	Commarea	Request IP Addr:			
TOR System ID:		AOR System ID:			
Trans ID:		TOR Task No:			
Elapsed (ms) >	0	Max Records:	900	Result Type:	DHTML View

Task Date	Task Time	Program Name	Method Name	Program Type	Requester IP
2014-05-09	18.01.08	SOLACA01	commareaTest	CA	10.5.20.39
2014-04-24	22.09.09	SOLACA01	QA1005M1	CA	10.5.20.39
2014-04-24	18.06.42	SOLACA01	QA12CAM1	CA	10.5.20.39
2014-04-24	16.42.56	SOLACA01	commareaTest	CA	10.5.20.116
2014-04-21	10.52.19	SOLACA01	QA1005M1	CA	10.5.20.39
2014-03-05	08.45.16	SOLACA01	QA1007M1	CA	10.5.20.35
2014-03-01	21.40.50	SOLACA01	QA11CM01	CA	10.5.20.35
2014-02-28	17.57.45	SOLACA01	QA10CM02	CA	10.5.20.35
2014-02-28	16.56.59	SOLACA01	QA10CM02	CA	10.5.20.35
2014-02-28	13.44.00	SOLACA01	QA10CM01	CA	10.5.20.35
2014-02-28	08.44.45	SOLACA01	QA1005M1	CA	10.5.20.35

The information is organized under a series of columns:

- **Task Date:** the day the transaction was generated, represented as yyyy-mm-dd. Clicking on the date for a specific transaction displays the search details panel that contains very detailed information about the transaction.



Task Date	Task Time
2008-06-19	07.12.42
2008-06-19	07.12.31
2008-06-19	07.10.44

- **Task Time:** the time the transaction was generated, represented as hh.mm.ss.

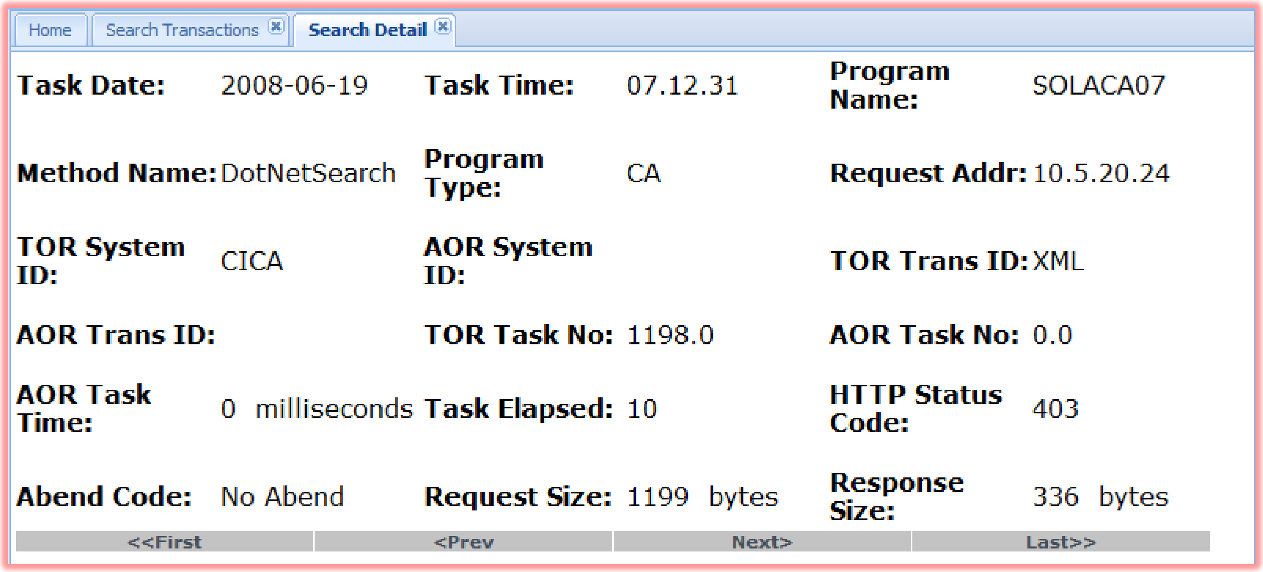
- **Program Name:** the program whose execution generated the transaction.

- **Method Name:** the name of the method whose execution generated the transaction.

- **Program Type:** the category (type) of program whose execution generated the transaction.

- **Requester IP:** the IP Address of the originating request (responsible for executing the method that generated the transaction, if it comes via HTTP).

To get detailed information about a specific transaction, click on the transaction date. This will display the search detail panel.



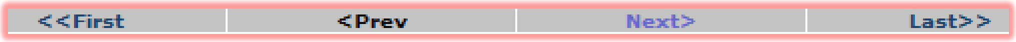
Task Date:	2008-06-19	**Task Time:**	07.12.31	**Program Name:**	SOLACA07
Method Name:	DotNetSearch	**Program Type:**	CA	**Request Addr:**	10.5.20.24
TOR System ID:	CICA	**AOR System ID:**		**TOR Trans ID:**	XML
AOR Trans ID:		**TOR Task No:**	1198.0	**AOR Task No:**	0.0
AOR Task Time:	0 milliseconds	**Task Elapsed:**	10	**HTTP Status Code:**	403
Abend Code:	No Abend	**Request Size:**	1199 bytes	**Response Size:**	336 bytes

This panel contains detailed information about a specific transaction organized under the following headings:

- **Task Date:** the date (yyyy-mm-dd) of the transaction.
- **Task Time:** the time (hh.mm.ss) of the transaction.
- **Program Name:** the program whose execution generated the transaction.
- **Method Name:** the method whose execution generated the transaction.
- **Program Type:** the type of the program whose execution generated the transaction.
- **Request Addr:** the IP Address of the originating request (responsible for executing the method that generated the transaction).
- **TOR System ID:** unique identifier for the TOR region where the transaction originated.
- **AOR System ID:** unique identifier for the AOR region where the transaction originated.
- **TOR Trans ID:** unique identifier given to each program that runs in a TOR.

- **AOR Trans ID:** unique identifier given to each program that runs in an AOR.
- **TOR Task No:** unique identifier that is given to each unique instance of a program running in a TOR.
- **AOR Task No:** unique identifier that is given to each unique instance of a program running in an AOR.
- **AOR Task Time:** how long it took to execute the program in the AOR, accurate to +/- 5 milliseconds.
- **Task Elapsed:** the total end to end time (AOR+TOR) that it took to execute the program, accurate to +/- 5 milliseconds.
- **HTTP Status Code:** the HTTP response code generated as a result of the transaction (e.g. 200 – OK, 403 – Auth Failure, etc.)
- **Abend Code:** the mainframe abend code if the program abnormally terminates (i.e. abnormally ends - abends).
- **Request Size:** the size of the input SOAP XML in bytes.
- **Response Size:** the size of the output SOAP XML in bytes.

The links at the bottom of the panel allow you to navigate through all the transactions in the list.



<<First: show details for the first transaction in the list.

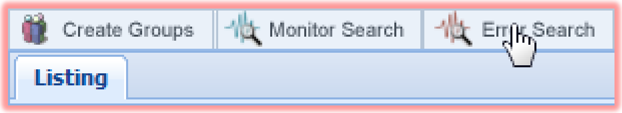
<Prev: show details for the previous transaction.

Next>: show details for the next transaction.

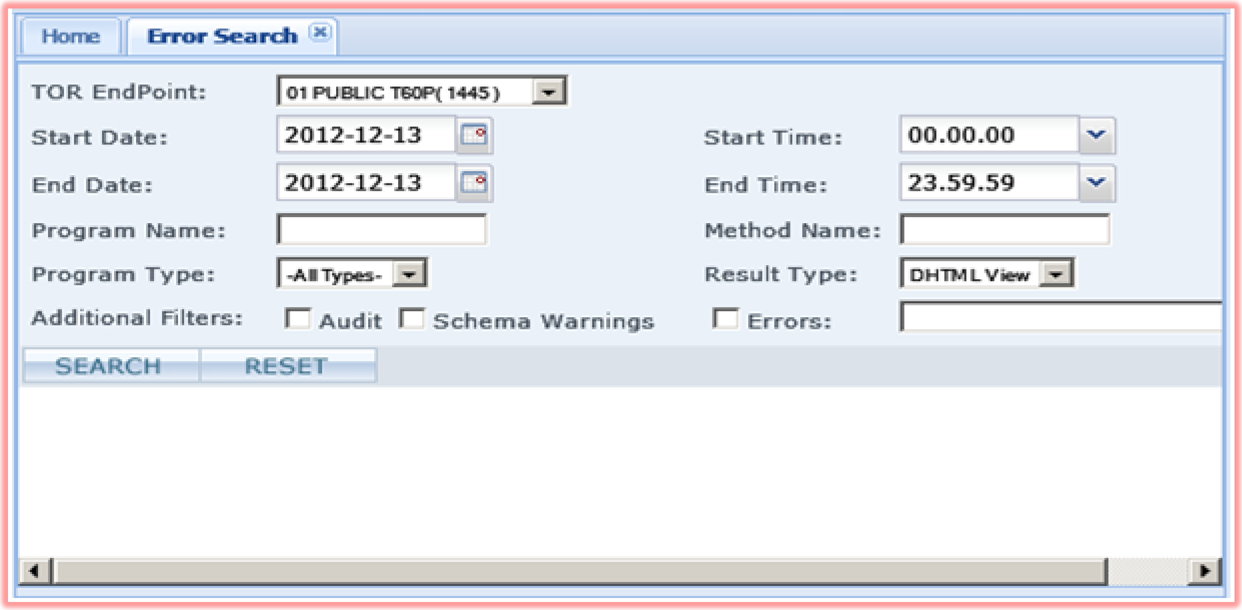
Last>>: show details for the last transaction.

Error Logs

You can search the error logs by clicking the **Error Search** button on the button bar.



This will display the error search panel.



TOR EndPoint: 01 PUBLIC T60P(1445)

Start Date: 2012-12-13

Start Time: 00.00.00

End Date: 2012-12-13

End Time: 23.59.59

Program Name:

Method Name:

Program Type: -All Types-

Result Type: DHTML View

Additional Filters: □ Audit □ Schema Warnings □ Errors:

To conduct a search of the error log, enter search parameters using the search fields to narrow the scope of your search. You can also conduct a search with the default (mostly blank) settings.

The following is a description of the search fields:

- **TOR EndPoint:** narrows the search to errors generated within a matching SOLA container.
- **Start Date and End Date:** the start and end dates are automatically populated with the current date, though these values can be changed if necessary. All errors are stamped with the date and time at which they take place, and only errors that took place on or after the start date and on or before the end date will be returned.
- **Start Time and End Time:** the start and end times are automatically populated with the current system time and can be changed by manually entering a time (hh.mm.ss). All errors are stamped with the date and time at which they take place, and only errors that took place at or after the start time and at or before the end time will be returned.
- **Program Name:** narrows the search to errors generated by the specified program.
- **Method Name:** narrows the search to errors generated by the specified method.

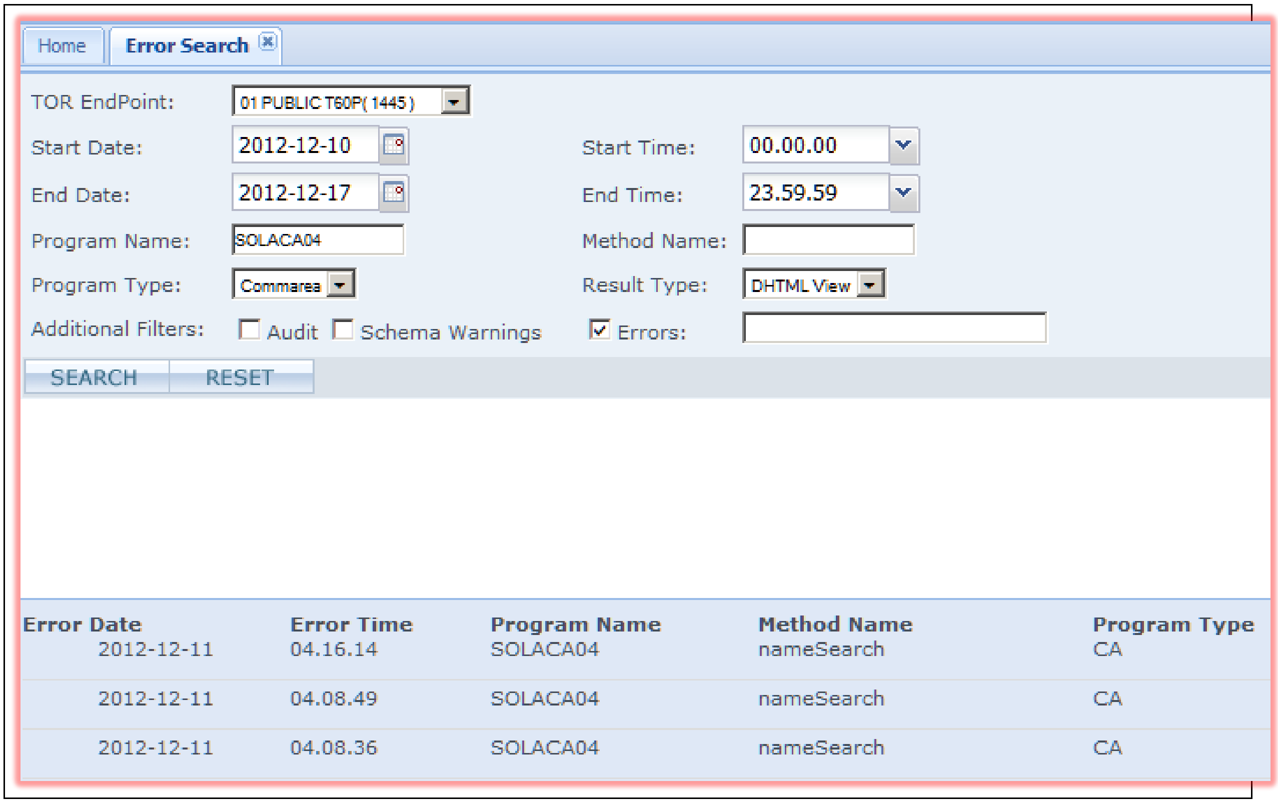
- **Program Type:** narrows the search to errors generated by a method executed by the specified program type. Options are All Types, Commarea, Callable, BMS3270, Outbound, AdhocSQL, TgadpXml or Custom.

- **Result Type:** specifies how the results will be displayed, either as html (normal view) or as an Excel spreadsheet. Selecting Excel will download the results and open MS Excel (if installed), displayed the data in an Excel spreadsheet.

- **Additional Filters:** narrows the search to include only Audit Information, Schema Warnings or specific Error codes.

Once you have specified your search parameters, click SEARCH.

The results of the search will be displayed below the error search panel. If the list exceeds the available screen size, then you will need to scroll to see all of the search results.



TOR EndPoint: 01 PUBLIC T60P(1445)

Start Date: 2012-12-10   Start Time: 00.00.00

End Date: 2012-12-17   End Time: 23.59.59

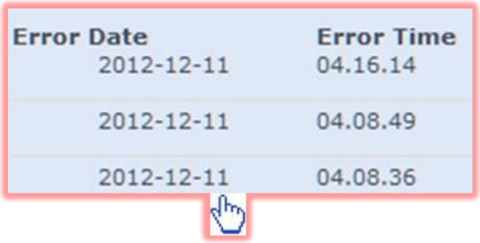
Program Name: SOLACA04   Method Name:

Program Type: Commarea   Result Type: DHTML View

Additional Filters: □ Audit □ Schema Warnings ☑ Errors:

Error Date	Error Time	Program Name	Method Name	Program Type
2012-12-11	04.16.14	SOLACA04	nameSearch	CA
2012-12-11	04.08.49	SOLACA04	nameSearch	CA
2012-12-11	04.08.36	SOLACA04	nameSearch	CA

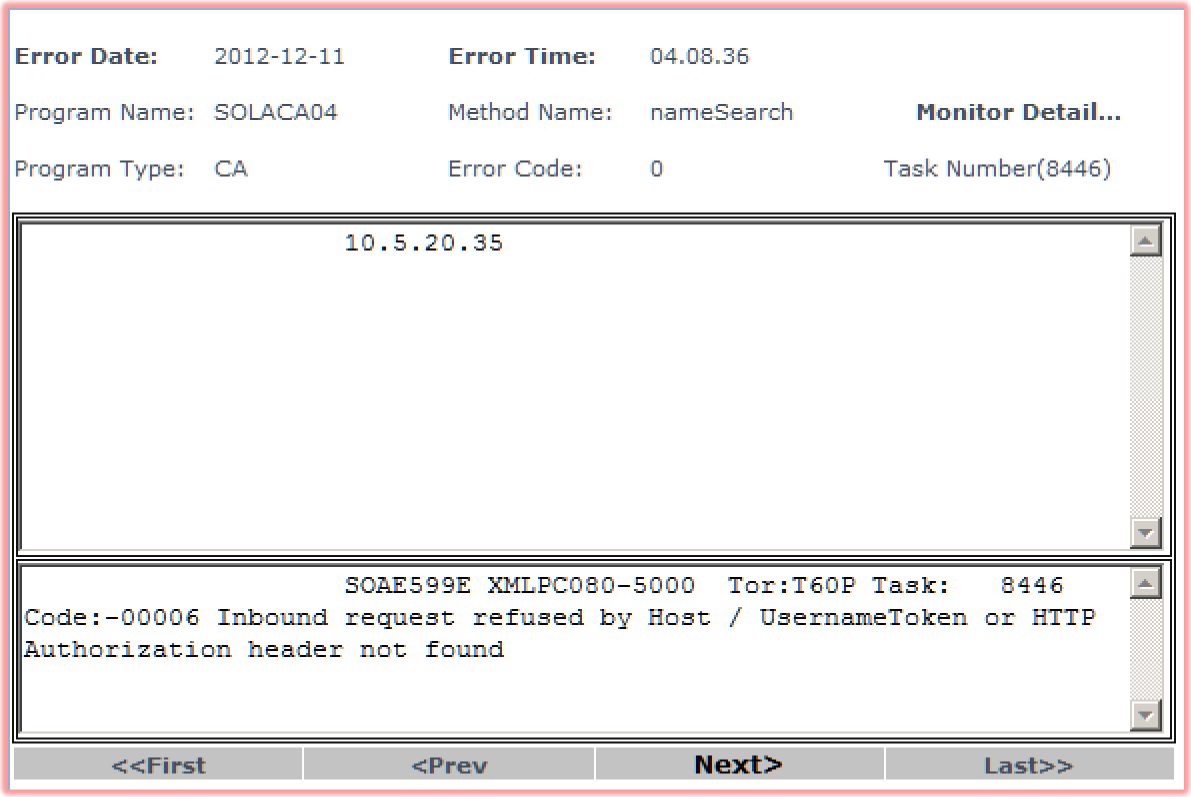
The information is organized under a series of columns:



Error Date	Error Time
2012-12-11	04.16.14
2012-12-11	04.08.49
2012-12-11	04.08.36

- **Error Date:** the day the error was generated, represented as yyyy-mm-dd. Clicking on the date for a specific error displays the search details panel that contains detailed information about the error.
- **Task Time:** the time the error was generated, represented as hh.mm.ss.
- **Program Name:** the program that generated the error.
- **Method Name:** the name of the method that generated the error.
- **Program Type:** the category (type) of program that generated the error.

To get detailed information about a specific error, click on the error date. This will display the search detail panel.



Error Date: 2012-12-11 **Error Time:** 04.08.36

Program Name: SOLACA04 Method Name: nameSearch **Monitor Detail...**

Program Type: CA Error Code: 0 Task Number(8446)

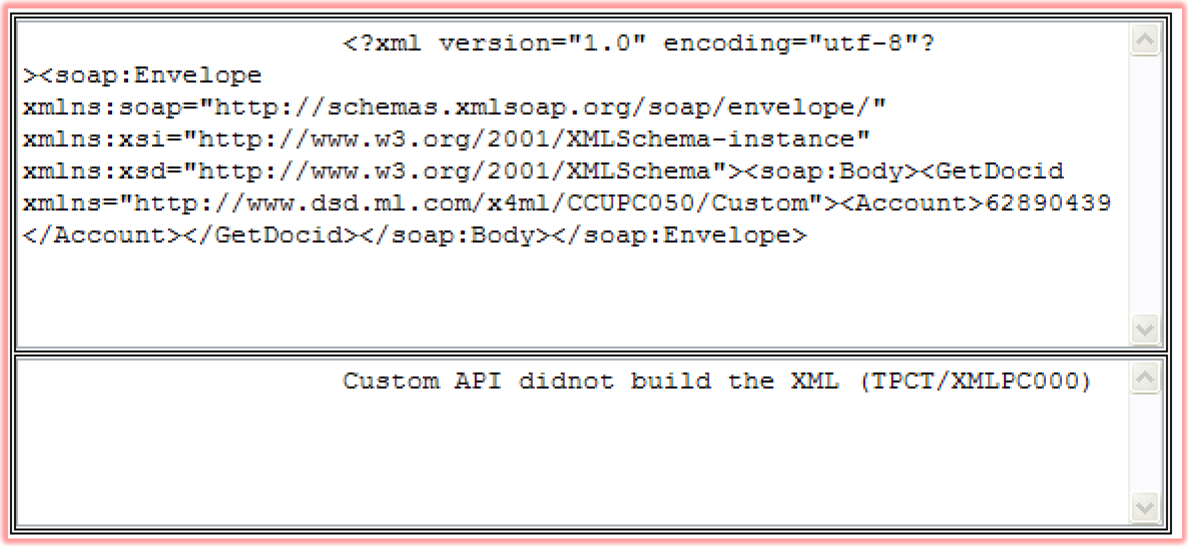
```
                          10.5.20.35
```

```
                          SOAE599E XMLPC080-5000  Tor:T60P Task:   8446
Code:-00006 Inbound request refused by Host / UsernameToken or HTTP
Authorization header not found
```

This panel contains detailed information about a specific error organized under the following headings:

- **Error Date:** the date (yyyy-mm-dd) of the error.
- **Error Time:** the time (hh.mm.ss) of the error.
- **Program Name:** the parent program of the method that caused the error.
- **Method Name:** the method that caused the error.
- **Program Type:** the type of program that caused the error.
- **Error Code:** the error code of the generated error.
- **Task Number:** the SOLA task number of the task that caused the error.

This panel may contain an error display field that contains additional debugging information.

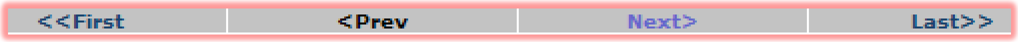


```
                     <?xml version="1.0" encoding="utf-8"?
><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetDocid
xmlns="http://www.dsd.ml.com/x4ml/CCUPC050/Custom"><Account>62890439
</Account></GetDocid></soap:Body></soap:Envelope>
```

```
                     Custom API didnot build the XML (TPCT/XMLPC000)
```

This field is divided into two panes. The bottom pane displays the mainframe error message, while the top pane displays the input XML that caused the error.

The links at the bottom of the panel allow you to navigate through all the errors in the list.



<<First: show details for the first error in the list.

<Prev: show details for the previous error.

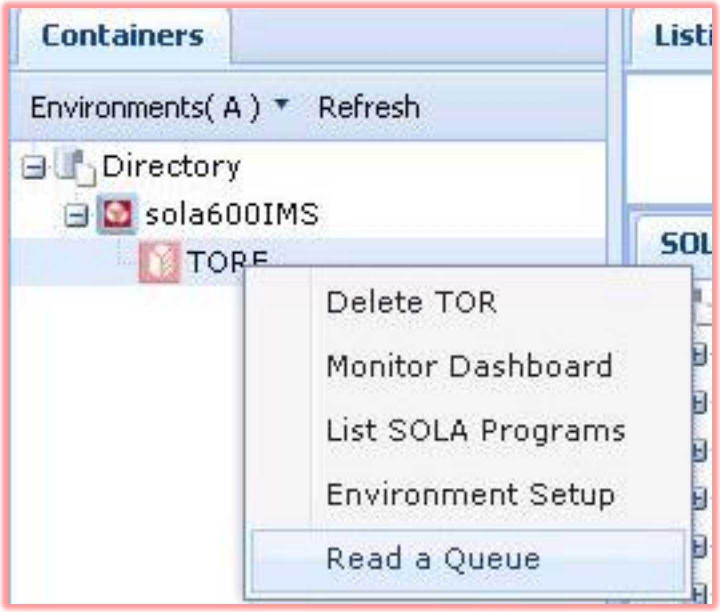
Next>: show details for the next error.

Last>>: show details for the last error.

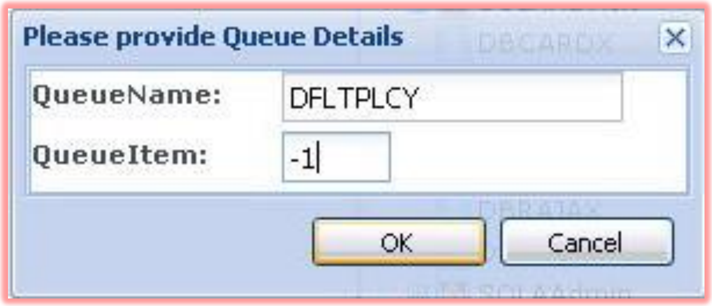
Reading Temporary Storage Queues

The SOLA Resource Manager provides the ability to read CICS Temporary Storage queues that reside in the SOLA CICS Container. SOLA uses TS queues extensively for caching and tracing purposes.

In the Containers window, right click on the SOLA Container and the following menu will be displayed.



Click on the "**Read a Queue**" option to display the queue selection pop-up window. Enter the queue name in QueueName field and the item number (record number) to read in the QueueItem field. If you specify -1 in QueueItem then all of the items in the queue will be read.

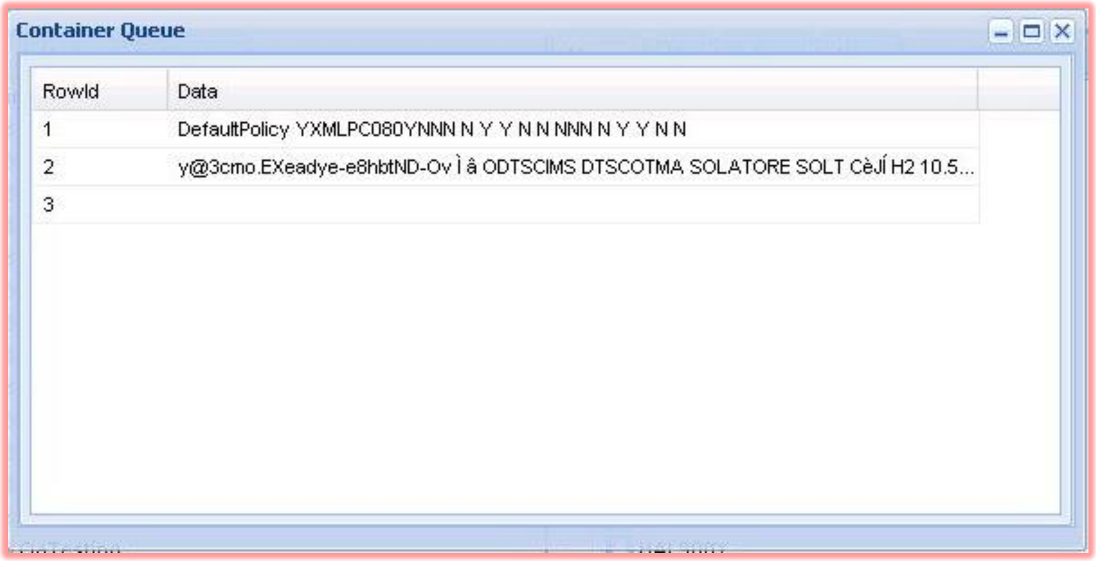


QueueName: DFLTPLCY

QueueItem: -1

In the example above we're reading all of the items (-1 in QueueItem) on the DFLTPLCY queue (SOLA's Default Policy queue).

SOLA will attempt to read the queue in the SOLA CICS container, and will format the results in a pop-up window, as shown below:



RowId	Data
1	DefaultPolicy YXMLPC080YNNN N Y Y N N NNN N Y Y N N
2	y@3cmo.EXeadye-e8hbtND-Ov Ì â ODTSCIMS DTSCOTMA SOLATORE SOLT CèJÍ H2 10.5...
3	

Result of reading DFLTPLCY queue in the SOLA CICS Container.

Identity Mapping

Each SOLA runtime container incorporates a comprehensive identity mapping subsystem that can be used to map credentials from one format to another. The intent of this subsystem is to provide single sign-on capability across an enterprise and including the mainframe.

The Identity Mapping subsystem consists of several components:

- The SOLA Analyzer XMLPCAN
- The Analyzer template XML#DAN
- The credentials mapping database
- A CICS Maintained Data Table containing the credentials mapping cache
- The Identity Mapping Panel in Resource Manger

Identity Mapping Setup

Before using SOLA's identity mapping subsystem you must perform some one-time setup steps. These consist of configuring the XMLPCAN analyzer and the XML#DAN template.

SOLA Analyzer XMLPCAN

Transactions that run through CICS Web Support use the default CICS analyzer DFHWBADX. If your transactions require Identity Mapping then you will need to use the SOLA Analyzer XMLPCAN. To use the SOLA Analyzer, define XMLPCAN in the URM parameter of the TPCPIPS service definition. CICS Web Support will then run the SOLA Analyzer on every web service request.

To change the analyzer, use CEDA transaction as follows:

```
OVERTYPE TO MODIFY                                      CICS RELEASE = 0640
 CEDA  ALter TCpipservice( TOREXT   )
  TCpipservice   : TOREXT
  GROup          : TORWEB
  DEscription  ==>
  Urm          ==> XMLPCAN
  POrtnumber   ==> 01743              1-65535
  STatus       ==> Open               Open | Closed
  PROtocol     ==> Http               Iiop | Http | Eci | User
  TRansaction  ==> CWXN
  Backlog      ==> 00005              0-32767
  TSqprefix    ==> X4ML
  Ipaddress    ==>
  SOcketclose  ==> 000010             No | 0-240000 (HHMMSS)
  Maxdatalen   ==> 005032             3-524288
 SECURITY
  SSl          ==> Yes                Yes | No | Clientauth
  CErtificate  ==> LABEL00000001
 +   (Mixed Case)

                                                  SYSID=TORE APPLID=TOREXT
```

If you want to try identity mapping without making permanent changes, you can update the TCPIPS service definition with the CEMT transaction as follows:

```
   CEMT I TCPIPS(*)
```

The following will be displayed:

```
   Tcpips(TOREXT  ) Ope Por(01743) Http Ssl Tra(CWXN)
       Con(00000) Bac( 00005 ) Max( 005032 ) Urm(DFHWBADX) Sup
```

You can update the URM parameter to XMLPCAN by overwriting the URM with XMLPCAN and pressing the Enter key. This update will be lost the next time that CICS is recycled.

Analyzer template XML#DAN

The XML#DAN template is located in SAMPLIB. In future releases XML#DAN will be maintained by Resource Manager; however in the current release you must configure it manually by updating it with ISPF Edit, assembling it into your DFHRPL library and issuing a newcopy.

XML#DAN consists of 3 sections:

1. A header section
2. A counter section
3. An XPath section

Header Section

The header section is shown below:

```
XML#DAN      CSECT
XML#DAN      AMODE 31
XML#DAN      RMODE ANY
          ENTRY XML#DAN   Template Name
          DC CL4'ANLZ'    EyeCatcher
          DC A(XPATHPTR) XPATH POINTER
MMOD      DC CL8'XMLPCMAP' Program to invoke map/auth
          DC CL256' ' Filler Area
```

The Header section contains the name of the Identity Mapping module, XMLPCMAP, which ships with SOLA. If you wish to use a different module, replace XMLPCMAP (shown in MMOD above) with the name of the module you want to use. Your module must be defined with EXECKEY(CICS).

If you don't want to use Identity Mapping then replace XMLPCMAP with spaces.

Counter Section

The counter section is shown below:

```
XPATHPTR  EQU *
MCTR      DC XL04'2' Nbr Xpath Expressions
```

Replace the counter, shown as MCTR, with the number of XPath expressions (as shown in the example below).

```
SUBType1  DC CL1'W'
          DC CL50'UsernameToken/Username                            '
          DC CL50'                                                  '
          DC CL50'                                                  '
          DC CL50'                                                  '
          DC CL50'                                                  '
          DC CL5'     '
SUBType2  DC CL1'S'
          DC CL50'Assertion/AuthenticationStatement/Subject/NameIden'
          DC CL50'tifier                                            '
          DC CL50'                                                  '
          DC CL50'                                                  '
          DC CL50'                                                  '
          DC CL5'     '
*---- List 255 bytes XPath expressions
          END XML#DAN
```

In the example above, there are two XPath expressions. The first yields a WS-Security UserName token and the second yields a Subject token.

The format of the XPATH expression is very simple; each entry is 256 bytes long and it consists of a 1 byte subtype and a 255 byte search string. There is no limit on the number of expressions.

The Subtype field has two possible values:

- W The XPath expression yields a WS-Security Username token
- S The XPath expression yields a subject Username token

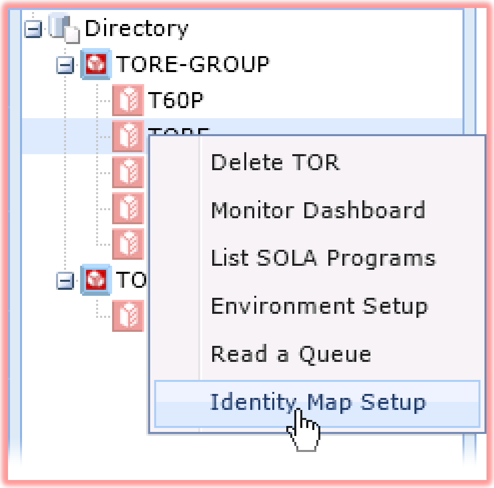
The SUBType1 value of "W" indicates that the XPath expression will yield a WS-Security UserName token. XMLPCAN expects that UserName token to be a RACF ID, and will limit the length of the token to 8 bytes. If you wish to map a longer UserName, for example an LDAP ID, you would define it as a Subtype S (for Subject).

The 255 bytes following contain the XPath expression are used to search the XML document for the Username token. In this example above we are searching for `'UsernameToken/Username'`.

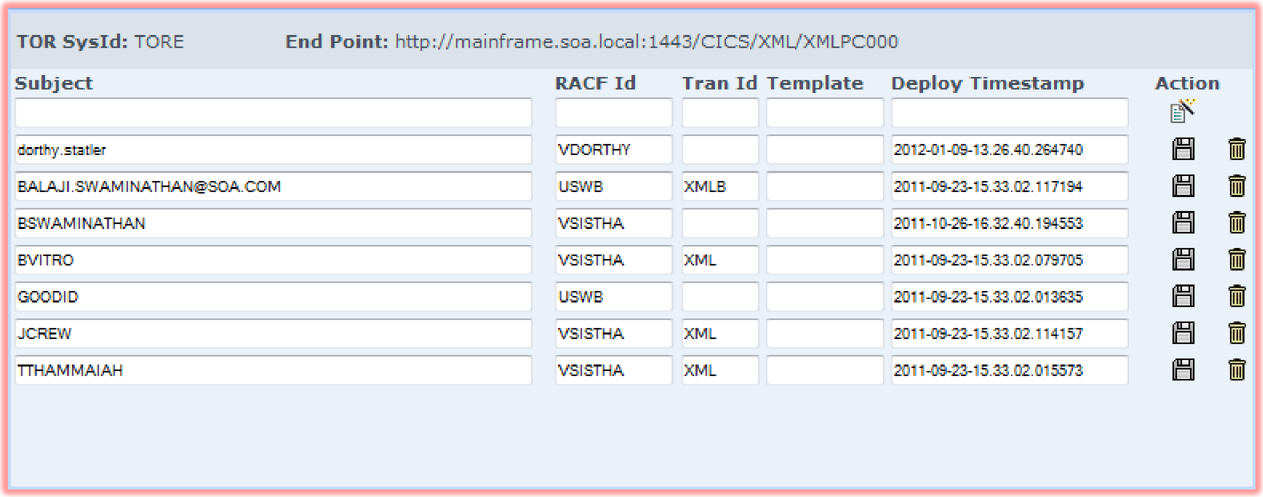
The SUBType2 value of "S" indicates that the XPath expression will yield a Subject token.

The 255 bytes following contain the XPath expression to be used to search the XML document for the Subject. In our case, we will be searching for `' 'Assertion/AuthenticationStatement/Subject/NameIdentifier'`.

Using Identify Mapping



In the Containers Panel, right click on a container and select **Identity Map Setup** from the pop-up menu. This will display the Identity Map Setup panel. All identity mappings created in this panel will only apply to the selected container.



TOR SysId: TORE **End Point:** http://mainframe.soa.local:1443/CICS/XML/XMLPC000

Subject	RACF Id	Tran Id	Template	Deploy Timestamp	Action	
dorthy.statler	VDORTHY			2012-01-09-13.26.40.264740		
BALAJI.SWAMINATHAN@SOA.COM	USWB	XMLB		2011-09-23-15.33.02.117194		
BSWAMINATHAN	VSISTHA			2011-10-26-16.32.40.194553		
BVITRO	VSISTHA	XML		2011-09-23-15.33.02.079705		
GOODID	USWB			2011-09-23-15.33.02.013635		
JCREW	VSISTHA	XML		2011-09-23-15.33.02.114157		
TTHAMMAIAH	VSISTHA	XML		2011-09-23-15.33.02.015573		

To create identity maps, fill in the information on the top(blank) row:

- **Subject:** this is the primary ID that you will use for single-sign-on. Other IDs will map to this one.
- **RACF Id:** use this field to specify a mainframe ID to map to the primary ID.
- **Tran Id:** this optional field can be used to override the transaction that SOLA runs under (the transaction is part of the authorization) If this field is left blank, the default transaction XML will be used.
- **Template:** this optional field can be used to override the identity mapping template XML#DAN. This field is reserved for future use. Do not populate this field in this version of SOLA.
- **Deploy Timestamp:** this is a system populated field (leave it blank) that indicates when the identity map was created.

Once you have filled in the required fields, click the 📝 icon to create the identity map.

To delete an existing identity map, click that identity map's corresponding 🗑 icon.

To make changes to an identity map, edit the fields you want to change, then click the identity map's corresponding 💾 icon.

Making Identity Mapping Changes Manually

If you do not want to use the Identity Mapping Panel in Resource Manager, you can make changes directly to identity mapping DB2 table, TBXMLMFD, which is laid out as follows:

- **TOR_SYS_ID:** char(4), SYSID of WOR region to which the identity map will apply. Identity mapping only applies to the one container.
- **ID_MAP_TYP:** char(1), type of mapping. This value must always be set to O for identity mapping.
- **SUBJECT_ID:** varchar(255),subject ID. This is the primary ID that you will use for single-sign-on. Other IDs will map to this one.
- **RACF_USR_ID:** char(8), RACF mainframe user id.   Use this to specify a mainframe ID to map to the primary ID.
- **TRN_ID:** transid , optional, overrides XML transaction. This can be used to override the transaction that SOLA runs under (the transaction is part of the authorization) If this field is left blank, the default transaction XML will be used.
- **TMPLT_ID:** template Id, optional, overrides identity mapping template XML#DAN. This field is reserved for future use. Do not populate this field in this version of SOLA.
- **DEPLOY_TS:** this is a system populated field (leave it blank) that indicates when the identity map was created.

Run the job IMAPLOAD after updating the input load cards with the mapping data. To force the SOLA run-time to use this updated data, purge the temporary storage queue 'XML8SOLAIDENTMAP' in the WOR region. Otherwise, it will only take effect after the region is recycled.

Writing Your Own Mapping Program

SOLA allows you to use your own identity mapping facility in place of XMLPCMAP.

Mapping is performed while running under the CICS Web Support transaction CWXN. If you have specified your own program name module in the XML#DAN template, then the SOLA Analyzer XMLPCAN will call your program with the following Commarea, after it has extracted the username and subject names from the input SOAP message:

```
01  DFHCOMMAREA.
  05  MF-Return-Code          PIC S9(04) BINARY.
      88  Mapping-Found       VALUE +0.
      88  Mapping-Not-Found   VALUE +1.
      88  No-Group-Id-Found   VALUE +2.
      88  Group-Not-Found     VALUE +3.
      88  Mapping-Error       VALUE -1.
      88  Mapping-Abend       VALUE -99.
  05  MF-Return-Msg           PIC  X(100).
  05  Mapping-Facility-Inputs.
      10  MF-Subject          PIC  X(254).
      10  MF-IP-Address.
          15  MF-IP-Node-1    PIC  X(01).
          15  MF-IP-Node-2    PIC  X(01).
          15  MF-IP-Node-3    PIC  X(01).
          15  MF-IP-Node-4    PIC  X(01).
      10  MF-User-Id-In       PIC  X(08).
      10  MF-Program-Name     PIC  X(08).
      10  MF-Method-Name      PIC  X(64).
  05  Mapping-Facility-Outputs.
      10  MF-User-Id          PIC  X(08).
      10  MF-Tran-Id          PIC  X(04).
  05  MF-LogErrors            PIC  X(1).
  05  MF-Filler               PIC  X(255).
```

You will be passed the following information:

1. The Subject, extracted from the input SOAP
2. The IP Address of the requestor
3. The Username token, extracted from the input SOAP
4. The name of the target Program
5. The web service operation (known by SOLA as MethodName)

You will be expected to return the following information:

1. RACF User ID to run the transaction under. If you return spaces then SOLA won't attempt to change the User ID.
2. CICS Transaction ID to run the transaction under. If you return spaces then SOLA won't attempt to change the Transaction ID.

The source for this copybook is shipped in the SOLA distribution SAMPLIB as MAPAREA.