

# Integration Framework of SAP Business One Settings for Development Environment

10, 2015



This course covers the recommended settings for using the integration framework as a development environment for creating new scenario packages. This course is only relevant if you plan to use the integration framework for scenario package development.

## Objectives

---

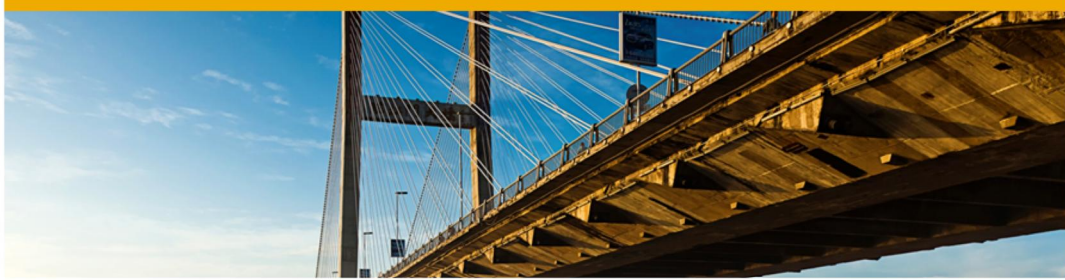


At the end of this course, you will be able to:

- Change the integration framework environment from a productive system to a development system
- Configure the message log for use in a development environment
- Further configure the development system by:
  - Setting up notifications for blocked queues
  - Changing the default error handling for scenario processing
  - Expanding Tomcat memory
  - Starting the integration framework in safe mode

After completing this topic, you will be able to:

- Change the integration framework environment from a productive system to a development system
- Configure the message log for use in a development environment
- Optionally make additional configurations in a development system by:
  - Setting up notifications for blocked queues,
  - Changing the default error handling for scenario processing
  - Expanding Tomcat memory
  - Starting the integration framework in safe mode



# System Profiles

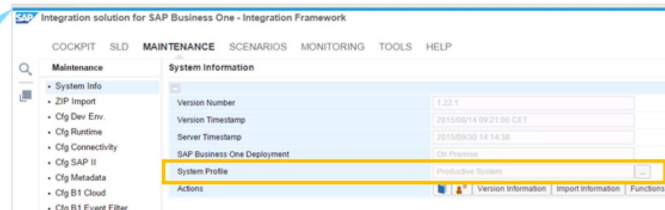


The first topic describes the different system profiles for productive and development environments.

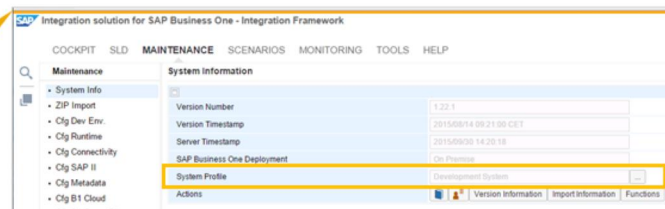
## Development and Productive System Profiles

- The integration framework can operate as either a productive system or a development system
- After installation, default is a productive system profile
- To create and test new scenarios, you need to switch the system profile to a development system

Productive system indicated by blue upper line



Development system indicated by gold upper line



The integration framework can operate in one of two modes – either as a productive system or as a development system. This is controlled by the system profile. A system profile brings together the recommended configuration settings in one place, optimized for a productive system or a development system.

Each mode has a color-coded upper line – blue for a productive system and yellow/gold for a development system.

After installation, the integration framework is set for a productive system profile. If you want to create and test new scenarios, you need to switch the system profile to a development system.

## Version and System Information

Path: *Maintenance* → *System Info*

Integration framework information:

- **Version Number** of installed integration framework
- **Version Information:** list of changes with each version
- **Functions:** available inbound and outbound channels, atoms and functions in current release

The screenshot displays the SAP Integration Framework user interface. The top navigation bar includes tabs for COCKPIT, SLD, MAINTENANCE, SCENARIOS, MONITORING, TOOLS, and HELP. The MAINTENANCE tab is active, and the left sidebar shows a tree structure with 'System Info' selected. The main area is titled 'System Information' and shows details for version 1.22.1, including the version number, timestamp, and server timestamp. Below this, there are three buttons: 'Version Information', 'Import Information', and 'Functions'. The 'Functions' button is highlighted with a red box. Below the main area, there are two pop-up windows. The left window, titled 'Integration Framework - System Information', shows a list of version changes with columns for Function Group, Version, and Description. The right window, also titled 'Integration Framework - System Information', shows a table of functions with columns for System, Mode, Origin, External, and Available.

System	Mode	Origin	External	Available
VORD	Asynchronous	Event	Valid	Available
Predecessor	Asynchronous	Event	Handled	Available
Internal Channel	Asynchronous	Event	Handled	Available
Web Service	Synchronous	Call	Handled	Available
Web Service	Asynchronous	Call	Handled	Not Yet Available
HTTP Call	Synchronous	Call	Handled	Available
HTTP Call	Asynchronous	Call	Handled	Not Yet Available
SAP BAPI	Asynchronous	Event	Handled	Available
Business One	Synchronous	Msg	DI Service	Not Yet Available
Business One	Synchronous	Msg	DI Object	Not Yet Available
Business One	Synchronous	Msg	SQL	Not Yet Available
Business One	Synchronous	Msg	Control	Not Yet Available
Business One	Synchronous	Msg	Multiple	Not Yet Available
Business One	Asynchronous	Event	DI Object	Available
Business One	Asynchronous	Event	DI Service	Available
Business One	Asynchronous	Event	SQL	Available
Business One	Asynchronous	Event	Control	Not Yet Available

© 2015 SAP SE or an SAP affiliate company. All rights reserved.

5

When you use the integration framework for scenario development, it is useful to know the capabilities and features available to you.

Select the *System info* function under the Maintenance tab of the integration framework user interface to see information about the installed version of the integration framework. If you need to report integration framework errors to SAP Support, you will be asked for the version number, which is shown in this screen.

By clicking the *Version Information* button, you can display a list of changes implemented in your current integration framework version.

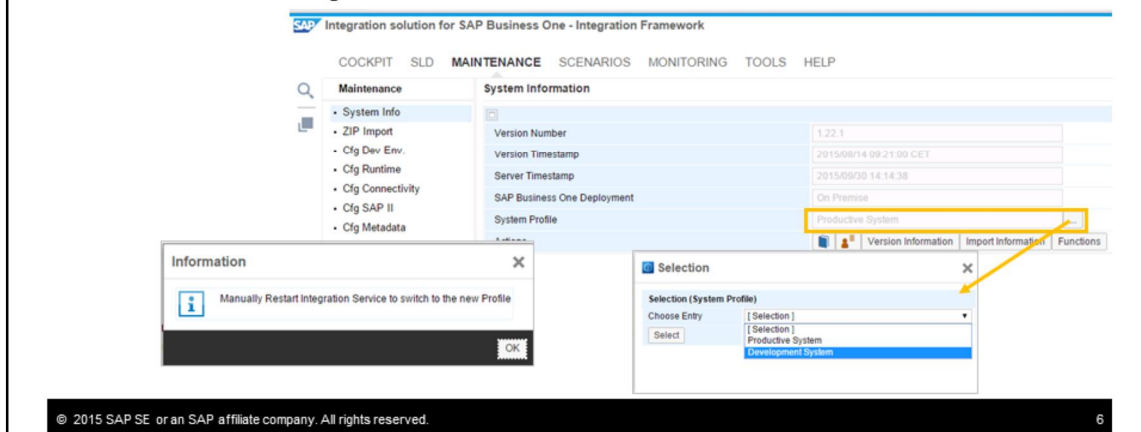
By selecting the *Functions* button you can view the scope of functionality available in your current integration framework version for building scenarios. For example, you can see the inbound and outbound channels, atoms, and features available to you for scenario development.

Note that the function list is increased with successive releases of the integration framework.

## Switching to the Development Profile

Path: *Maintenance* → *System Info*

- The development system profile contains the recommended configuration settings for scenario development, testing and debugging.
- To switch to the development system profile, select the ellipsis button for the *System Profile* field and choose *Development System* Then restart the *SAP Business One Integration Service*



As of SAP business One release 9.1 PL09 and later releases, the integration framework allows you to easily switch over to a development system profile so you can create and test new scenario packages.

The profile contains the recommended configuration settings for scenario development, testing and debugging. Prior to this release, in order to configure a development environment, you needed to edit and change several settings in the *xCellerator* configuration file. Now all these settings are gathered together in one place in the system profile.

Switching profiles is done from the *Maintenance* tab. Choose *System Info*, then select the ellipsis button to the right of the *System Profile* field and choose *Development System*.

The system will prompt you to restart the *SAP Business One Integration Service* for the change to take effect.

Follow the same process to switch back to a productive system.

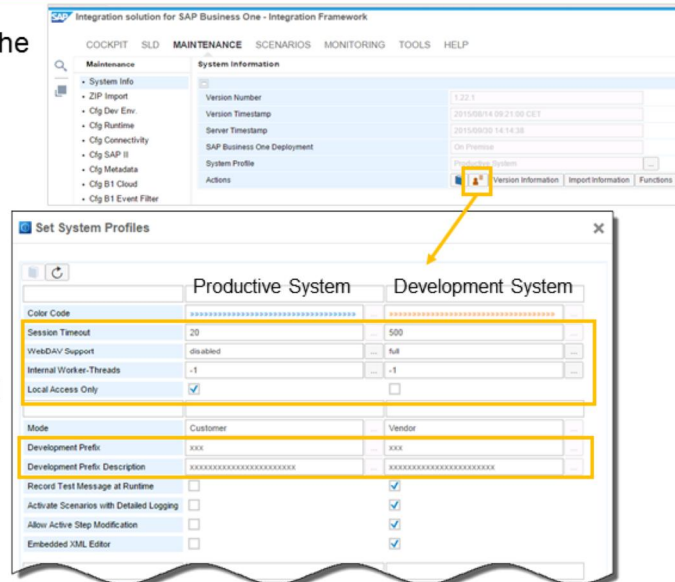
The settings specific to the development profile are discussed next.



## Development System Profile

Path: *Maintenance* → *System Info*

- Choose the icon to open the System Profiles
- The development system profile is optimized for scenario development:
  - The session timeout is increased
  - WebDAV support is enabled
  - Remote access enabled
- You should enter a development prefix and description for your company



© 2015 SAP SE or an SAP affiliate company. All rights reserved.

7

To open the system profiles, choose the icon located to the right of the documentation icon in the *System Info* screen. You will see both system profiles so you can compare the settings side by side. The active system profile is the one selected in the System Profile field.

While you can edit any of the values in the development system profile, the values are optimized for scenario development:

- The session timeout value is increased for login sessions, to avoid log outs
- WebDAV support is enabled (set to full). This enables a WebDAV client, such as an XML editor, to access the BizStore, allowing you to edit the XSL documents of the transformation atoms while developing and testing scenarios. For security reasons, this parameter is disabled in a productive system.
- Remote access is enabled, allowing you to access the integration framework from a remote computer. Obviously in a productive system this setting is disabled and access is restricted to local for security reasons.

When you first use the development system profile you should enter a development prefix and description for your company. This prefix is added to all scenarios you develop and identifies your content within the integration framework. SAP recommends using a three character prefix.

Note: The number of internal threads is set to default (-1). If you use more than one processor, you should change this parameter to reflect the number of available processors. You should also change this in the productive system profile to take advantage of multiple processors.

## Development System Profile - 2

Settings in the development profile for testing and debugging scenarios:

- Record test messages at runtime for later use
- Activate scenarios with detailed logging for step-by-step process flow debugging
- Allow modification of an activated step during testing
- Embedded XML editor

	Productive System	Development System
Color Code	.....	.....
Session Timeout	20	500
WebDAV Support	disabled	full
Internal Worker-Threads	-1	-1
Local Access Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Mode	Customer	Vendor
Development Prefix	xxx	xxx
Development Prefix Description	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Record Test Message at Runtime	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Activate Scenarios with Detailed Logging	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Allow Active Step Modification	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Embedded XML Editor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Log Level	InfoSet	Full message
Message Log	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B1 Event Monitor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Enforce Message Log for Simplified Calls	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Final Message Log Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Asynchronous Transaction - Processing	Retrial after 1 minute and stop processing of t	Retrial after 1 minute and stop processing of t
xCellerator cfg Log Level	SEVERE	CONFIG
Scenario Debug Info	<input type="checkbox"/>	<input checked="" type="checkbox"/>

The development profile has settings enabled for testing and debugging of new scenarios:

- The option to record test messages at runtime. When this is enabled, and you trigger a scenario step for testing, the input XML message will be automatically saved in the scenario step folder of the BizStore. You can later re-use this message for further testing or debugging without the need to connect to sender systems.
- The option to activate scenarios with detailed logging. To use this feature, you must first enable the capture of detailed logging in the transaction log by setting the xCellerator log level in the development system profile. Then, when a scenario is activated the detailed information will be automatically shown when you debug a scenario step.
- The option to allow modification of an active step. This is useful for testing new scenarios, as you can edit and change an activated scenario without having to deactivate it first.
- The embedded XML editor. This XML editor is provided by SAP and does not need a license. During scenario development and testing, if you enable this, you can open and edit XML files directly in transformation atoms and for format control documents.



## Development System Profile - 3

Settings for detailed logging of messages:

- Message log level is *Full message*
- *B1 Event Monitor* switched on
- Message log entries enforced for HTTP and Web service calls
- Logged messages can be updated during processing (Final message only checkbox unchecked)

	Productive System	Development System
Color Code	.....	.....
Session Timeout	20	500
WebDAV Support	disabled	full
Internal Worker-Threads	-1	-1
Local Access Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Mode	Customer	Vendor
Development Prefix	xxx	xxx
Development Prefix Description	.....	.....
Record Test Message at Runtime	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Activate Scenarios with Detailed Logging	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Allow Active Step Modification	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Embedded XML Editor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Log Level	InfoSet	Full message
Message Log	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B1 Event Monitor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Enforce Message Log for Simplified Calls	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Final Message Log Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Asynchronous Transaction - Processing	Retrial after 1 minute and stop processing off	Retrial after 1 minute and stop processing off
Accelerator cty Log Level	SEVERE	CONFIG
Scenario Debug Info	<input type="checkbox"/>	<input checked="" type="checkbox"/>

The development profile has settings enabled for the message log:

- The level of information written to the message log is set to *Full message*, meaning that logged messages include the XML message details. For a productive system this is an overhead.
- The B1 event monitor is enabled for a development system, and shows events originating in SAP Business One.
- The checkbox is enabled to enforce message log entries for simplified HTTP and Web service calls, even if the settings have been disabled in the call.
- The final message only checkbox is unchecked allowing the logged message to be updated during integration framework processing. In a productive system only the final message is logged, for performance reasons.

Note that many of these message log settings can also be set from the *Maintenance > Cfg MsgLog* screen.

## Development System Profile - 4

Settings for capture of detailed transaction information:

- xCerator log level set to CONFIG
- Scenario debug enabled

	Productive System	Development System
Color Code	.....	.....
Session Timeout	20	500
WebDAV Support	disabled	full
Internal Worker-Threads	-1	-1
Local Access Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Mode	Customer	Vendor
Development Prefix	xxx	xxx
Development Prefix Description	.....	.....
Record Test Message at Runtime	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Activate Scenarios with Detailed Logging	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Allow Active Step Modification	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Embedded XML Editor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Log Level	InfoSet	Full message
Message Log	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
B1 Event Monitor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Enforce Message Log for Simplified Calls	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Final Message Log Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Asynchronous Transaction - Processing	Retrial after 1 minute and stop processing of f	Retrial after 1 minute and stop processing of f
xcerator cty Log Level	SEVERE	CONFIG
Scenario Debug Info	<input type="checkbox"/>	<input checked="" type="checkbox"/>

The development system profile has settings to enable the capture of detailed transaction information for debugging:

- The *xCerator* manages all processes and adapter connections in the integration framework, and the log level parameters specify how much information is captured during processing. The CONFIG level is the most detailed level. For a productive environment the log level should be set to SEVERE to minimize logging overhead.
- The scenario debug option enables the display of debugging arrows in the atom processing flow as you test and debug scenario steps. Note that you can also control this setting from the *Process Control* section of the *Monitoring* tab.

For more information on detailed debugging, see the companion course "*Introduction to Scenario Monitoring and Debugging*."

## Additional Message Log Configuration -1-

Path: *Maintenance* → *Cfg MsgLog*

To avoid large log files, make sure daily deletion is enabled

- **Daily Deletion:** Enable automatic daily deletion of message logs
- **Indexed Access:** improves performance of accessing message logs (default)
- **Blocksize:** Number of messages displayed
- **User Interface Refresh Timeout:** controls how long time filters are displayed in message log view

The screenshot shows the SAP Maintenance configuration interface. The left sidebar lists various configuration areas, with 'Cfg MsgLog' selected. The main panel is titled 'Message Log Handling' and contains several settings:

Configuration Item	Value / Status
Message Log	<input checked="" type="checkbox"/>
Enforce Message Log for Simplified Calls	<input checked="" type="checkbox"/>
Final Message Log Only	<input type="checkbox"/>
Daily Deletion	<input checked="" type="checkbox"/>
Indexed Access	<input checked="" type="checkbox"/>
Block Size	50
Default View	Grouped
User Interface Refresh Timeout	0
Log Level	
Actions	

At the bottom right, there are buttons for 'Manual Deletion', 'Daily Deletion', and 'Filter'.

© 2015 SAP SE or an SAP affiliate company. All rights reserved.

11

To avoid large log files, you should set up automatic deletion of the message logs. You do this from the *Cfg MsgLog* option of the *Maintenance* tab:

- Select the **Daily Deletion** checkbox so that the integration framework automatically deletes message log entries on a regular basis.
- **Indexed Access** improves the performance of accessing message logs. This is the default setting. If you switch on indexed access, you will see additional options for viewing the message log information in the message log monitor.
- **Blocksize.** Here you can enter the number of messages displayed in the message log list on one page. The default block size is 50.
- **User Interface Refresh Timeout** value. This value controls for how long the message log screen remembers the timestamps which are used as filters in the Message Log view. The default is 10 minutes. If you enter 0, when you view the Message Log from the path *Monitoring > Message Log*, the user interface opens with the default time filters to display the whole day.

Note: The Message Log checkbox will appear checked here, based on the development system profile.

## Additional Message Log Configuration -2-

Path: *Maintenance* → *Cfg MsgLog* – *Daily Deletion*

Message Log Handling

Message Log	<input checked="" type="checkbox"/>
Enforce Message Log for Simplified Calls	<input checked="" type="checkbox"/>
Final Message Log Only	<input type="checkbox"/>
Daily Deletion	<input checked="" type="checkbox"/>
Indexed Access	<input checked="" type="checkbox"/>
Block Size	50
Default View	Grouped
User Interface Refresh Timeout	0
Log Level	Full message
Actions	<input type="checkbox"/> Manual Deletion <input checked="" type="checkbox"/> Daily Deletion <input type="checkbox"/> Filter

Message Log - Automatic Deletion

Deletion Time (Hour)	23
Backup Buffer (Days)	5
Last Deletion	2015.05.12 23:00
Delete Imported Messages	<input type="checkbox"/>
Actions	<input type="button" value="Close"/>

Parameters for daily deletion of logs:

- **Deletion Time:** hour
- **Backup Buffer:** number of days to exclude from deletion
- **Delete Imported Messages:** imported messages included in daily deletion

Option to instantly delete log by selecting *Manual Deletion* checkbox

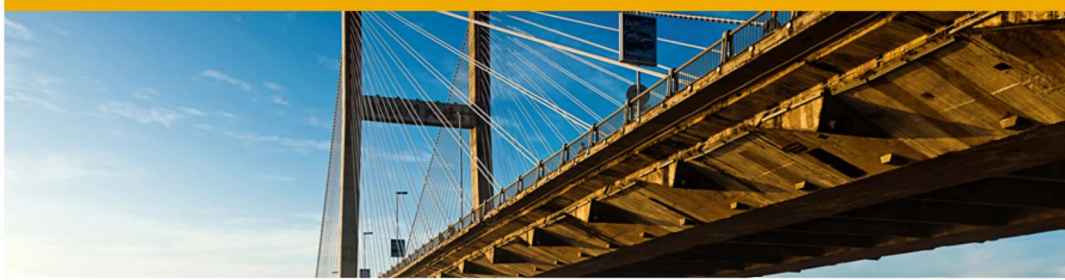
If you enable the *Daily Deletion* checkbox, you should set parameters for the daily deletion run by selecting the *Daily Deletion* button:

- The *Deletion Time* value is the hour when the logs will be deleted each day (the default time is 11 pm, or 23 hours). It is important to set the time as the log files can get large.
- The *Backup Buffer* is the number of days the integration framework keeps the messages until the next deletion (the default is 5 days).
- If you check the *Delete Imported Messages* checkbox, imported messages will be included in the daily deletion run. An example of importing messages is if you are supporting a customer, and need more details about a certain situation. You can ask the customer to export message log information from the message log monitor and send it to you. You can then import this information into your message log monitor for analysis, as if it had been produced in your local integration framework. The imported log contains all detailed debug information if enabled in the customer system.

To save these settings, click the *Save* button.

Note: The *Last Deletion* field shows the timestamp of the last automatic deletion.

You also have the option to instantly delete log information one time by selecting the *Manual Deletion* button to the left of the *Daily Deletion* button.



# Error Handling



As well as the system profiles, you can specify error handling options for your development environment.

## Maintenance

### Default Error Handling

Path: *Maintenance* → *Cfg Error Handling*

- View the default error handling for synchronous and asynchronous transaction processing
- Select the error handling option for Processing phase

Transaction	Action
Synchronous Transaction	Call fails with http error or soap fault
Synchronous Transaction - Inbound	No action or retrial after 1 minute, if possible
Synchronous Transaction - Processing	Retrial after 1 minute and stop processing of following messages
Synchronous Transaction - Splitter	Retrial after 1 minute and stop processing of following messages
Synchronous Transaction - Outbound	Remove from queue and put to error inbox

Actions: Overall Actions | Outbound Error Actions | Garbage Collection

You can also define error handling in the scenario package. Choose *Scenarios > Package Design*, and select the *Definitions* button.

© 2015 SAP SE or an SAP affiliate company. All rights reserved.

14

For developing new scenarios, it is helpful to know how the integration framework performs error handling.

Choose the path *Maintenance > Cfg Error Handling* to view the default actions for error handling. In all cases a failure message is written to the message log.

For synchronous transactions, the call fails with a fault and a notification goes to the sender system.

For asynchronous transactions, the error handling depends on the phase. The default actions are shown in the slide. For the processing phase, you have an option to either to retry after one minute and stop processing (as shown in the slide) or to remove from queue and continue processing.

In addition to the default error handling shown here, the integration framework offers the options to define specific error handling steps on scenario package level. To define the scenario-specific error handling, choose *Scenarios -> Package Design*, click the *Definitions* button and select the *Error Handling* option.



## Maintenance

### Error Handling – Overall Actions

Path: *Maintenance* → *Cfg Error Handling* → *Overall Actions*

- Choose the *Overall Actions* button to enable a notification to be sent when a message blocks an internal queue
- Choose the *Configure* button to provide the email details

The screenshot shows the SAP configuration interface for error handling. It starts with the 'Default Error Handling' table, which lists various transaction types and their default actions. The 'Overall Actions' button is highlighted. Below this, the 'Overall Error Actions' dialog is shown, where the 'QBlocker Alert' checkbox is checked and the 'QBlocker Alert Frequency' is set to '1,10'. The 'Configure' button is highlighted. Finally, the 'QBlock Alert' configuration dialog is shown, where the 'Sender E-mail Address', 'Receiver E-mail Address', and 'Subject' fields are visible, along with the 'Attach B11 Message' checkbox and the 'SMTP' action button.


The *Overall Actions* button allows you to configure a notification e-mail to be sent if a message blocks an internal queue for a long time. The notification is important since all other messages in the queue behind this message cannot be processed and they stay in the queue. To do this:

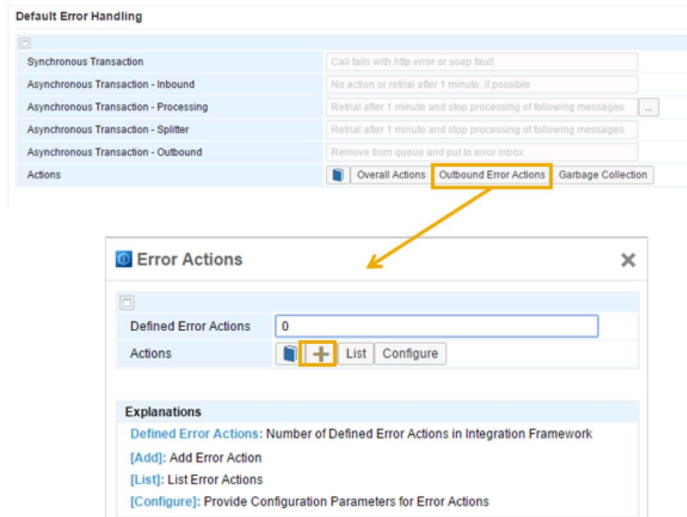
- Choose the *Overall Actions* button and enable the *QBlocker Alert* checkbox.
- Then define how many times you want to receive a notification e-mail. If you enter 1,10, for example, you receive an e-mail when a message blocks the queue for the first time, and after the tenth attempt of the integration framework to process the message again.
- Save your settings.
- Then, choose the *Configure* button, to provide the details for the notification e-mail, such as sender and receiver e-mail address, and e-mail subject. You can choose to attach the erroneous integration framework message to the e-mail.

## Maintenance


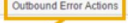

### Error Handling – Outbound Error Actions - 1

Path: *Maintenance* → *Cfg Error Handling* → *Outbound Error Actions*

- Option to define specific error actions that apply at package, step, or outbound channel level
- Choose *Outbound Error Actions* button
- Choose  to define a new error action




The screenshot displays the SAP configuration interface for error handling. The top section, 'Default Error Handling', is a table with columns for transaction types and their corresponding actions. The 'Outbound Error Actions' button is highlighted in the 'Actions' row. Below this, the 'Error Actions' dialog box is open, showing a 'Defined Error Actions' counter at 0. The 'Add Error Action' (+) button is highlighted in the 'Actions' section. The dialog also includes a 'List' button and a 'Configure' button. An arrow points from the highlighted button in the table to the dialog box.

Default Error Handling	
Synchronous Transaction	Call fails with http error or soap fault
Asynchronous Transaction - Inbound	No action or retry after 1 minute, if possible
Asynchronous Transaction - Processing	Retry after 1 minute and stop processing of following messages
Asynchronous Transaction - Splitter	Retry after 1 minute and stop processing of following messages
Asynchronous Transaction - Outbound	Remove from queue and put to error inbox
Actions	  

**Error Actions**

Defined Error Actions: 0

Actions:  List Configure

**Explanations**

Defined Error Actions: Number of Defined Error Actions in Integration Framework

[Add]: Add Error Action

[List]: List Error Actions

[Configure]: Provide Configuration Parameters for Error Actions

The *Outbound Error Actions* button allows you to define specific error actions that the integration framework performs if errors occur during the outbound phase.

These error actions can apply at the scenario package, scenario step, or outbound channel level. To define outbound error actions, choose the *Outbound Error Actions* button then choose the *Add Error Action* (+) icon.

## Maintenance

### Error Handling – Outbound Error Actions - 2

Path: *Maintenance* → *Cfg Error Handling* → *Outbound Error Actions*

To define the error actions:

1. Select the error action type (scenario, step, or outbound)
2. Select the event that triggers the error action
3. Click the *Select Actions* button then select applicable actions to respond to the error action type and event
4. Choose the *Activate* button to activate the action

The screenshot displays the SAP 'Error Actions' configuration window. The 'Error Action Type' is set to 'Outbound Channel' and the 'Error Action Event' is 'SAP Business One'. The 'Select Actions' button is highlighted. An arrow points to the 'Integration Framework - Select Error Actions' window, which shows a list of actions: 'Keep in Inbound Queue', 'Automatic Retrial', 'Send E-Mail' (checked), 'Don't Send To Error Inbox', and 'Wait And Retry' (checked with a timer of 5).

© 2015 SAP SE or an SAP affiliate company. All rights reserved.

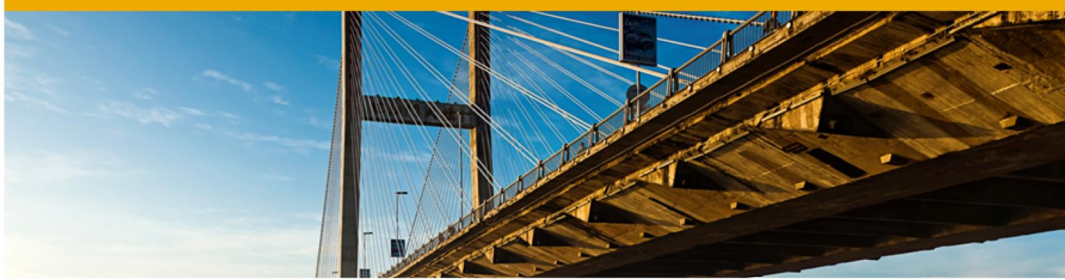
17

To define error actions from the Error Actions window:

1. Select the *Error Action Type* from the list to specify whether the error action applies to the scenario package, scenario step or outbound channel.
2. Select the *Error Action Event* from the list. The list shown depends on what you selected as the error action type. In this example, the SAP business One event is selected.
3. Choose the *Select Actions* button. In the Select Error Actions window select the applicable actions should an error occur in the error action type and event.
4. After you have defined the error action, remember to activate it.

The actions you can choose from are:

- **Keep in Inbound Queue:** The integration framework will keep the message in the inbound queue for outbound processing.
- **Automatic Retrial:** The integration framework will try to process the message again. You can provide timer settings in the *Configure* function of the *Error Actions* user interface.
- **Send E-Mail:** You can provide e-mail settings in the *Configure* function of the *Error Actions* user interface.
- **Don't Send to Error Inbox:** The integration framework will not send the message to the error inbox. Together with the *Keep in Inbound Queue* option, you keep the message in the integration framework and define an individual error handling.
- **Wait and Retry:** The integration framework will wait for a defined period of time and try to process the message again for a defined number of times. Only after this time, the integration framework performs the error handling and all other actions you have selected.



## Other Development Settings



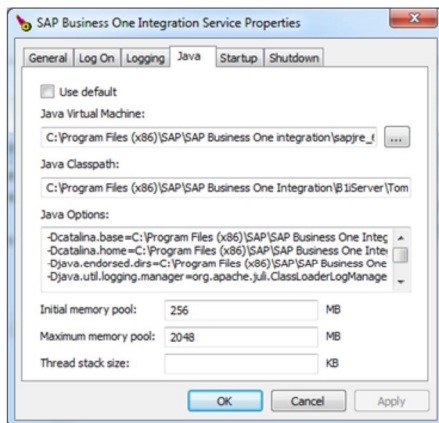
As well as the configuration settings, there are various maintenance settings for the development environment.

## Configuration

### Tomcat Memory

- Assign more RAM to improve performance:

Path: ..\SAP\SAP Business One\SAP Business One Integration\  
\Integration Server\Tomcat\bin\_64 (or bin)\tomcat6w.exe



- To assign Tomcat memory:
  - Open *tomcat6w.exe*
  - Select the *Java* tab
- Recommended values for a 32bit system:
  - Initial memory pool: 256 MB
  - Maximum memory pool: 1024MB
- Recommended values for a 64bit system:
  - Initial memory pool: 256MB
  - Maximum memory pool: 2048MB or more, depends on server memory, no limit defined by Tomcat

If you expect your development system to run under very high load and to process a huge amount of messages, you can assign more random-access memory (RAM) to the integration framework server to improve performance. You assign the memory at the Tomcat level.

To assign memory:

- Open the file *tomcat6w.exe* from the Tomcat folder. If the system denies access, select *tomcat6w.exe*, open the context menu and select the *Run as Administrator* option.
- Select the Java tab, and increase the maximum memory pool amount as follows:
  - In a 32-bit operating system, the maximum memory pool is 1024 MB.
  - In a 64-bit operating system, there is no maximum memory pool defined by Tomcat. The default is 2048 MB but you can increase it according to the server memory.

For more information on Tomcat memory, see the Operations chapter in the integration framework online help.

## Safe Mode Configuration

### xcellerator.cfg File

- You can start the integration framework in safe mode by changing the parameter in the configuration file:

```
Path: ..\SAP\SAP Business One\SAP Business One Integration\  
      \Integration Server\Tomcat\webapps\BliXcellerator\  
      \xcellerator.cfg
```

#### Safe Mode Use cases

- Locked administrator user – see SAP Note # 1556041
- An error-prone process blocks the integration framework, even after restart
- Integration framework will not start due to application deployment problems

#### Activation

- Set `xcl.safemode=true`
- Restart the integration service

#### Safe Mode Limitations

- All adapters are disabled
- User authentication is disabled
- Access is only possible from the local machine.

Similar to operating systems, you can start the integration framework in safe mode. Safe mode is not required to use the integration framework for development, but you should consider use safe mode in the following cases:

- You have locked the administrator user. This is probably the most common use case. For more information, see the SAP Note referenced in the slide. In safe mode the relevant administration tools still work, and you can unlock or reset the password for an administrator account.
- An error-prone process blocks the integration framework, even after a restart.
- The integration framework will not start due to application deployment problems.

To activate safe mode:

1. Edit the `xcellerator.cfg` file using the path shown in this slide, and change the value of `xcl.safemode` from 'false' to 'true'
2. Restart the SAP Business One Integration service then verify that the integration framework starts in safe mode.

In safe mode the integration framework adapters are disabled, user authentication is disabled, and access is possible only from the local machine. You should not run scenario packages in safe mode.

To reverse safe mode, edit the `xcellerator.cfg` file again and change the `safemode` parameter to false. Then restart the SAP Business One Integration service.



## Summary

- The integration framework provides system profiles for a development and productive environment. These profiles contain the recommended configuration settings for each environment
- After installation, the integration framework is set to a productive environment. To develop and test scenarios, you need to switch to the development system profile
- Development profile is optimized for scenario testing and debugging, including:
  - WebDAV client access
  - Remote access
  - Modification of active scenario step information
  - Full message logging to enable scenario monitoring
  - Detailed transaction logging and scenario debugging
- You can optionally set email notifications for blocked queues and select actions for outbound channel errors
- You have the possibility to expand Tomcat memory and start the integration framework in safe mode

- The integration framework provides system profiles for a development and productive environment. These profiles contain the recommended configuration settings for each environment
- After installation, the integration framework is set to a productive environment. To develop and test scenarios, you need to switch to the development system profile
- The development profile is optimized for scenario testing and debugging:
  - WebDAV client access
  - Remote access
  - Modification of active scenario step information
  - Full message logging to enable scenario monitoring
  - Detailed transaction logging and scenario debugging enabled
- In addition to the development profile, you can optionally set email notifications for blocked queues and select actions for outbound channel errors
- You have the possibility to expand Tomcat memory and start the integration framework in safe mode



**Thank You!**

**© 2015 SAP SE or an SAP affiliate company.  
All rights reserved.**

---

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. Please see <http://global12.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors.

National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.