

# Integration Framework for SAP Business One Log Maintenance in Integration Framework

PUBLIC



Global Roll-Out

January 2019, Nidhi Singh

THE BEST RUN





## TABLE OF CONTENTS

1.	SYSTEM PROFILES IN INTEGRATION FRAMEWORK .....	3
1.1.	Productive System Profile .....	3
1.2.	Development System Profile .....	3
1.3.	Development System Profile .....	4
1.4.	When to Activate Detailed Logging or to Choose 'Development Profile' .....	7
2.	ANALYZE INTEGRATION FRAMEWORK DATABASE SIZE .....	7
2.1.	Log Count Analysis in Integration Framework Database.....	7
2.2.	Check Daily Log Deletion / Garbage Collection .....	8
3.	INTEGRATION FRAMEWORK DATABASE CLEANUP .....	10
3.1.	Fast Deletion .....	10
3.2.	Slow Deletion .....	10

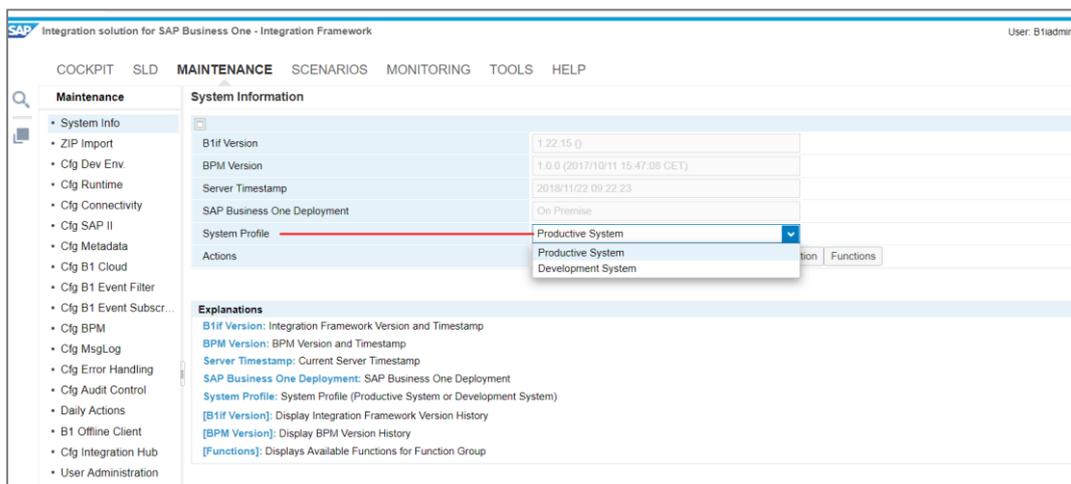
The integration framework for SAP Business One supports various log types to analyse and troubleshoot the daily transactions generated by your integration scenarios. It also facilitates the option of different system profiles like *Production* or *Development*. It is important to understand when to use the profiles, and the impact on the system based on the settings for the profile. This document focuses on the different log settings of the integration framework version 1 and gives advice about recommended settings.

## 1. SYSTEM PROFILES IN INTEGRATION FRAMEWORK

The integration framework for SAP Business One can be configured as a *Productive* system or a *Development* system in the *System Profiles*. Each profile contains different configuration settings for each environment. The profiles allow a quick configuration of different options in one convenient place. After login, you can easily change the *system profiles* in the *Maintenance* menu. The integration framework requires you to restart the integration service after changing the profile.

Note that the color of the upper line of the integration framework user interface is blue for a production system and yellow for a development system.

*Integration Framework for SAP Business One* → *Maintenance* → *System Info*



### 1.1. Productive System Profile

After completing the installation of the integration framework, the default profile is *Productive System*. This profile provides the recommended configuration required to work in a productive environment. The profile does not create detailed logs and keeps your productive system running smoothly. When running the integration framework in a live, productive environment, we recommend selecting the *Productive System* profile. For details about the standard *Productive System* configuration, see below.

### 1.2. Development System Profile

We set the *System Profile* to *Development System* when working in a development environment, for example, to create or update scenario steps. The profile provides the recommended configuration settings to edit, test and debug scenarios.

In this system profile, the Integration framework can create a large amount of log files that can affect the overall system performance and increase the size of the integration framework database.

We can occasionally use this option to troubleshoot issues in production environments as listed in section 1.4 of this document.



WebDAV client support is by default disabled in both the *Productive* and *Development* profile. We recommend changing the setting to *Full* in the *Development* profile, allowing easy access to the integration framework BizStore.

#### 1.3.4. Local Access Only Setting Details

Local Access Only	<input checked="" type="checkbox"/>	<input type="checkbox"/>
-------------------	-------------------------------------	--------------------------

For the *Productive* profile, the parameter is checked by default. The setting ensures that access to integration framework user interface is limited to the local machine (the server where the integration framework is installed). For the *Development* profile, the value is not selected by default. As such, the remote access to the integration framework user interface is possible from any server (assuming your network configuration also allows access).

#### 1.3.5. Development Environment (Mode) Setting Details

Development Environment	Profile PRODUCTIVE ENVIRONMENT	Profile DEVELOPMENT ENVIRONMENT
Mode	Customer	Vendor

The mode defines the namespace for scenario packages that you develop. If you are a scenario vendor, select *Vendor*. If you are a customer and you develop your scenario package, select *Customer*. Customers use the *z* namespace.

#### 1.3.6. Development Prefix Setting Details

Enter the development prefix for scenario development to be used in each profile.

#### 1.3.7. Activate Scenarios with Detailed Logging Setting Details

Activate Scenarios with Detailed Logging	0 - Switch off Debugging	1 - Switch on Step Debugging
--	--------------------------	------------------------------

For the *Productive* profile, detailed logging for scenarios is switched off. For the *Development* profile, step debugging is enabled for debugging the scenario steps. Step debugging allows debugging of the processing phase of a scenario step. To also allow debugging of *Inbound* and *Outbound*, choose *Full Debugging*.

#### 1.3.8. Record Test Message at Runtime Setting Details

Record Test Message at Runtime	<input type="checkbox"/>	<input checked="" type="checkbox"/>
--------------------------------	--------------------------	-------------------------------------

The setting is not selected in the productive profile. In the development profile, the setting is selected by default allowing the system to store test messages for each transaction in the *Design Home Directory* of the scenario step. Activating the recording of test messages in a productive system has a significant impact on performance.

#### 1.3.9. Allow Active Step Modification Setting Details

Allow Active Step Modification	<input type="checkbox"/>	<input checked="" type="checkbox"/>
--------------------------------	--------------------------	-------------------------------------

By default, the setting is selected only in the *Development* profile allowing you to make changes for an active scenario step while working in design phase.

#### 1.3.10. Embedded XML Editor Setting Details

Embedded XML Editor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
---------------------	--------------------------	-------------------------------------

Error Handling	Profile PRODUCTIVE ENVIRONMENT	Profile DEVELOPMENT ENVIRONMENT
Asynchronous Transaction - Processing	Retrial after 1 minute and stop processing c ▾	Retrial after 1 minute and stop processing c ▾

As of integration framework version 1.21.0, the integration framework provides an internal embedded XML editor. When this option is selected, the integration framework opens the internal XML editor when clicking the *xform* atoms (XSL documents). This setting is by default enabled in the *development* profile.

### 1.3.11. Log Level Settings Details

Message Log	Profile PRODUCTIVE ENVIRONMENT	Profile DEVELOPMENT ENVIRONMENT
Log Level	Infoset ▾	Full message ▾

Define the level of information the integration framework generates for the message log. The following levels are available

- Infoset
- Full message
- Full message incl. binary data

For a productive environment, we recommend switching off the message log or only switching it on with the lowest level (Infoset). If you develop scenario steps, switch on the full message log (Full message).

### 1.3.12. Message Log Setting Details

Message Log	<input type="checkbox"/>	<input checked="" type="checkbox"/>
-------------	--------------------------	-------------------------------------

By default, *Message Log* is not active for the *Productive* profile and we recommend switching it on only if you want to generate messages at scenario runtime.

### 1.3.13. B1 Event Monitor Settings Details

B1 Event Monitor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
------------------	--------------------------	-------------------------------------

If you want to display the events log in the integration framework for the events provided by SAP Business One Company databases, then switch on the *B1 Event Monitor*. This option is by default enabled in the *Development* profile.

### 1.3.14. Asynchronous Transaction Processing Setting Details

If an error occurs in the processing phase, the integration framework writes an entry to the *Failure* section of the message log. You have the following options to set the default behavior of the integration framework:

- Retrial after one minute and stop processing of following messages
- Remove from queue and continue with next message.

The integration framework writes messages removed from queues to an internal integration framework table.

For both *Productive* and *Development* profiles, the default setting is *Retrial after 1 minute and stop processing of following messages*.

For more details about this setting, check section 4(d) of SAP note [2192737 Best practices: Performance optimization of the Integration Framework](#).

### 1.3.15. Xcellerator.cfg Log Level Setting Details

Error Handling	Profile PRODUCTIVE ENVIRONMENT	Profile DEVELOPMENT ENVIRONMENT
Asynchronous Transaction - Processing	Retrial after 1 minute and stop processing c ▾	Retrial after 1 minute and stop processing c ▾

For a productive environment the *SEVERE* level is recommended and the default. To set a higher level is not advisable. In a development environment use the *CONFIG* level (default) that is required for creating debugging information.

### 1.4. When to Activate Detailed Logging or to Choose 'Development Profile'

- Developing, editing, testing and debugging scenarios.
- Before logging an incident with SAP Support, it is good to temporarily activate detailed logging, reproduce the issue, export the relevant message logs, and deactivate detailed logging again. These logs can then be provided via the Support incident.

#### For Example:

During processing of your scenario package, you see a failed message, in the *integration framework* → *Monitoring* → *Message log* with the following error:

**DI Error: (-5002) 10000100 - Document total value must be zero or greater than zero-The di call #1 did fail.**

To investigate this type of error, enable the detailed logging as described above. It captures the detailed information for an occurrence of the issue. This information can help to identify the root cause of an error.

#### Detailed Logging Impact on Integration Server Performance:

We do not recommend enabling detailed logging for an extended period, because it generates large log files for each transaction. With detailed logging enabled, the integration framework database can grow rapidly.

A large integration framework database can impact the system performance.

## 2. ANALYZE INTEGRATION FRAMEWORK DATABASE SIZE

### 2.1. Log Count Analysis in Integration Framework Database

If the integration framework database consumes a log of disk space, check the number of logs per dataset that indicates the collected data.

```
SELECT COUNT (*) AS "Total", BZSTIDXH.DATASET FROM "IFSERV"."BZSTIDXH", "IFSERV"."BZSTIDXP"  
WHERE
```

BZSTIDXH.PKEYGUID=BZSTIDXP.FKEYGUID GROUP BY BZSTIDXH.DATASET ORDER BY "Total" DESC

	Total	DATASET
1	7,224	com.sap.b1i.system.xc.iodata
2	2,368	com.sap.b1i.vplatform.scenarios.design
3	1,932	com.sap.b1i.vplatform.ide
4	952	com.sap.b1i.vplatform.doc
5	477	com.sap.b1i.vplatform.ui
6	422	com.sap.b1i.system.cc
7	373	com.sap.b1i.vplatform.runtime
8	297	com.sap.b1i.dev.ide
9	273	com.sap.b1i.system.xc
10	267	com.sap.b1i.system.iae
11	201	com.sap.b1i.system.uiframework
12	192	com.sap.b1i.b1imessage.logs
13	137	com.sap.b1i.system.sld.repository
14	113	com.sap.b1i.vplatform.bpm
15	112	com.sap.b1i.vplatform.repository
16	110	com.sap.b1i.vplatform.system
17	102	com.sap.b1i.hub.service
18	81	com.sap.b1i.system.sld.directory

Above, we can see that highest number of records is from the Transaction IO dataset (*com.sap.b1i.system.xc.iodata*).

It indicates that we should review our system profile/general logging configuration or check for transactions that are frequently retried. See more details in section 6 of SAP Note [1873607](#).

### Other Datasets with High Count

*com.sap.b1i.system.b1eventlog* – indicates that the B1 event monitor logging is enabled. If debugging is not required, turn off the logging.

*com.sap.b1i.system.b1imessage.logs* – indicates that message logs are enabled. Review the *Log Level* and the *Daily Deletion* configuration in the *integration framework* → *Maintenance* → *Cfg MsgLog*

*com.cxs.b1i.log* – indicates that the intercompany integration solution message logs are collected. Review the *Manage Logs* configuration in the *Intercompany Integration Administration Console* → *Preferences* → *Manager Logs*

## 2.2. Check Daily Log Deletion / Garbage Collection

The integration framework provides the garbage collection (GC) feature that automatically deletes logs to keep the integration framework database small. The daily deletion of message logs and garbage collection for error handling logs are enabled by default. **Configure the garbage collection (GC) for message logs:** Please make sure that you defined the correct configuration for daily deletion. Refer to the example below to check, if the daily deletion is working correctly:

## Example

Go to *integration framework* → *Maintenance* → *Cfg MsgLog*.

Deletion Time (Hour)	23	...
Backup Buffer (Days)	5	
Last Deletion	2018.10.15 23:00	
Delete Imported Messages	<input type="checkbox"/>	
Actions	Close	

**Explanations**

**Deletion Time (Hour):** Time for Daily Automatic Deletion  
**Backup Buffer (Days):** Buffer of Days Without Deletion  
**Last Deletion:** Date of Last Automatic Deletion  
**Delete Imported Messages:** Select to Let Garbage Collection Also Delete Imported Messages  
**[Save]:** Save

In the example shows that the last deletion of messages was performed on 15/Oct/2018. We configured 5 buffer days. With this configuration, any message created within the last 5 days is not deleted, all other messages are deleted.

You could also configure the garbage collection for error information-related messages in the integration framework at *Maintenance* → *Cfg Error Handling* to start the automatic deletion of error message logs.

To adjust the timer for garbage collection, please refer to section 2 of SAP Note [1873607](https://www.sap.com/support/1873607).

### 3. INTEGRATION FRAMEWORK DATABASE CLEANUP

If the logging levels were set incorrectly, some cleanup may be required in the database.

We can use the *Log Mass Deletion* tool for this activity allowing the option to delete B1 event logs, message logs and debug logs.

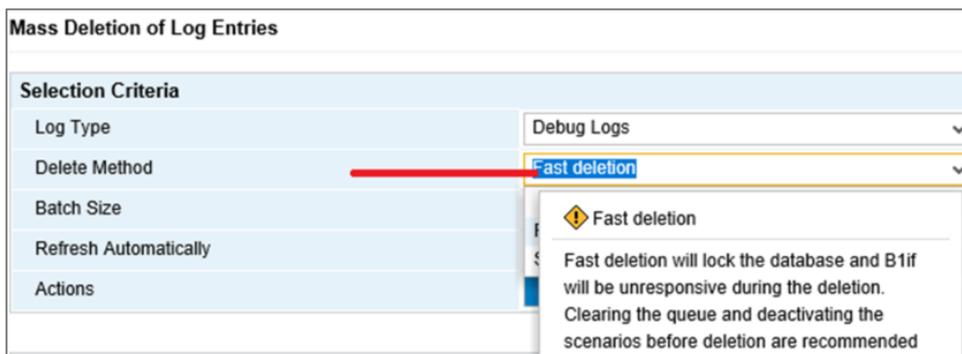
To use the tool, go to *integration framework* → *Tools* → *Log Mass Deletion*

Select the relevant logs for deletion. Selecting *Debug Logs* enables an additional option to avoid potential impact on the system performance. There are two different 'deletion methods' to delete the debug logs.

#### 3.1. Fast Deletion

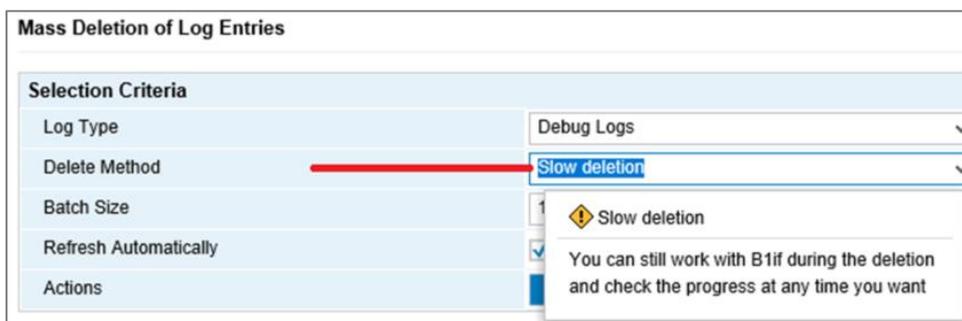
The fast deletion method is same as described above. The deletion of debug logs is faster, but the integration framework is unresponsive during the operation. We recommend performing the below mentioned steps, if you choose the fast deletion method:

- Deactivate the scenarios at *integration framework* → *Scenarios* → *Control*
- Ensure that the queues are empty at *integration framework* → *Monitoring* → *Queue Monitor*



#### 3.2. Slow Deletion

If you choose the slow deletion method, there is no impact on your integration scenarios, because the method performs the slow deletion of debug logs while working with the integration framework.



Please refer to SAP Note [2660491](#) for more details.

[www.sap.com/contactsap](http://www.sap.com/contactsap)

© 2018 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.

The information contained herein may be changed without prior notice. 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 or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP 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, and they should not be relied upon in making purchasing decisions.

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. All other product and service names mentioned are the trademarks of their respective companies. See <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.