



# Batch processing

Planon Software Suite

Version: L105

© 1997 - 2024 Planon. All rights reserved.

Planon and the Planon logo are registered trademarks of Planon Software Development B.V. or its affiliates. All other product and company names mentioned herein are trademarks or registered trademarks of their respective companies. Planon Software Development B.V., its affiliates and/or licensors own the copyright to all Planon software and its associated data files and user manuals.

Although every effort has been made to ensure this document and the Planon software are accurate, complete and up to date at the time of writing, Planon Software Development B.V. does not accept liability for the consequences of any misinterpretations, errors or omissions.

A customer is authorized to use the Planon software and its associated data files and user manuals within the terms and conditions of the license agreement between customer and the respective legal Planon entity as soon as the respective Planon entity has received due payment for the software license.

Planon Software Development B.V. strictly prohibits the copying of its software, data files, user manuals and training material. However, customers are authorized to make a back-up copy of the original CD-ROMs supplied, which can then be used in the event of data loss or corruption.

No part of this document may be reproduced in any form for any purpose (including photocopying, copying onto microfilm, or storing in any medium by electronic means) without the prior written permission of Planon Software Development B.V. No copies of this document may be published, distributed, or made available to third parties, whether by paper, electronic or other means without Planon Software Development B.V.'s prior written permission.

# About this Document

## Intended Audience

This document is intended for *Planon Software Suite* users.

## Contacting us

If you have any comments or questions regarding this document, please send them to: [support@planonsoftware.com](mailto:support@planonsoftware.com).

## Document Conventions

### **Bold**

Names of menus, options, tabs, fields and buttons are displayed in bold type.

### *Italic text*

Application names are displayed in italics.

### CAPITALS

Names of keys are displayed in upper case.

## Special symbols

	Text preceded by this symbol references additional information or a tip.
	Text preceded by this symbol is intended to alert users about consequences if they carry out a particular action in Planon.

# Table of Contents

Introduction.....	5
Process overview.....	5
Key requirements taken into account.....	6
Working with.....	7
Job definition.....	7
Job instance.....	7
Job execution.....	8
Step execution.....	8
Index.....	10

# Introduction

For processing large quantities of data that would otherwise consume too many resources or might cause performance issues, Planon features **Batch processing**.


**Batch processing** is bulk-oriented, non-interactive and runs in the background. It is long running, may be data- or computationally intensive and can be executed sequentially or in parallel. **Batch processing** can be initiated through various invocation models, such as: ad-hoc, scheduled, and on-demand.

## Examples

Typical examples of batch processing include (amongst others) processing of:

- Financial bookings
- Cost settlement calculations
- PPM order generation
- Order cleanup (fixing corrupt data or cleaning/archiving data or documents)

## Implementation


 **Batch processing** is a generic platform feature and the actual implementation requires a technical configuration. Currently, this feature is only available to internal Planon developers. Instructions for this implementation are available in the internal  *javadoc* .

- Because of its generic framework, Batch processing can be configured/implemented for many purposes.
- Similar to  *background actions* , the processing is done in the background, without the user "being aware" of it happening. The only difference is that **Batch processing** has a UI component (for the application manager) and can be configured to display progress on the process.

## Process overview

The following overview depicts the batch processing components in context.

- The **Execute batch step** is at the center of the process. This is the actual execution of the work that must be done. To optimize execution, the batch process must be *prepared* and after the work is done, it should be *finished*.

 What exactly must be done in these phases depends on the actual implementation and can hardly be generalized.

- During the execution of the process, the *prepare*, *execute* and *finish* phases have access to a **batch process and step context** in which data can be shared to optimize the execution.
- The **batch process control** and the **batch logging** are generic components:
  - The **control** component gives users the possibility to control the process.
  - The **batch logging** logs the progress and informs the users about the status of the batch process.

The batch process can be started in different ways, via an action, by a scheduler, or by any other trigger. It again depends on specific functionality, but it is always asynchronous. This means that when started manually, the user will only get a message such as 'The process has started'. To control the batch process, a specialized GUI (TSI step) is available to inspect the batch process.

## Key requirements taken into account

Batch processing is handled at framework level and is specifically designed to handle a large quantity of data. By design, it takes into account the following key requirements:

- Acceptable performance
- Efficient batch size
- Preventing/helping to solve errors
- Viewing progress
- Starting/stopping/restarting/abandoning the batch processes
- Scheduling
- File handling

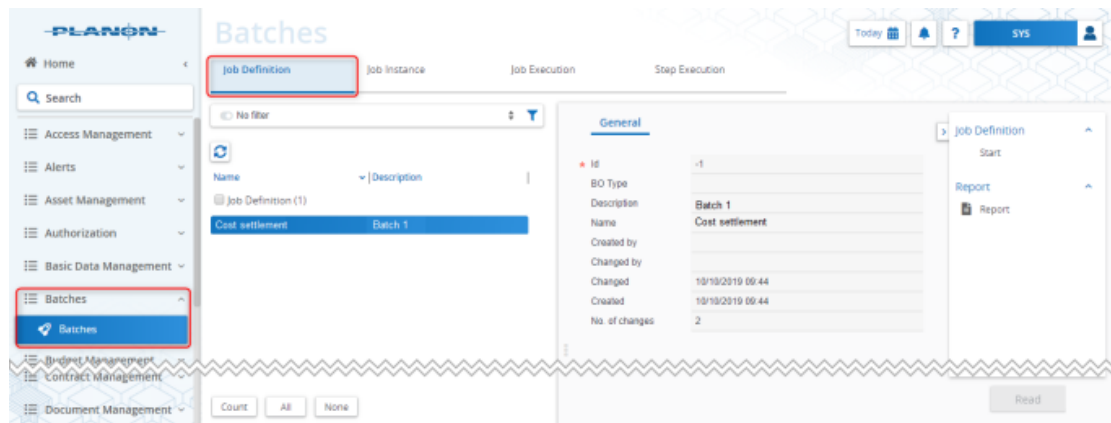
# Working with...

This section describes the various functions available.


## Job definition

Batch jobs are registered with the framework and are either started by an action or by a scheduled task.

If a batch is started for the first time, an entry is added to the **Job definitions** level. It displays the name and description of the batch job.



You can start batch jobs that do not require any batch job parameters by clicking **Start** on the action panel. If a batch job has batch job parameters, it requires its own action (BOM) and corresponding arguments.

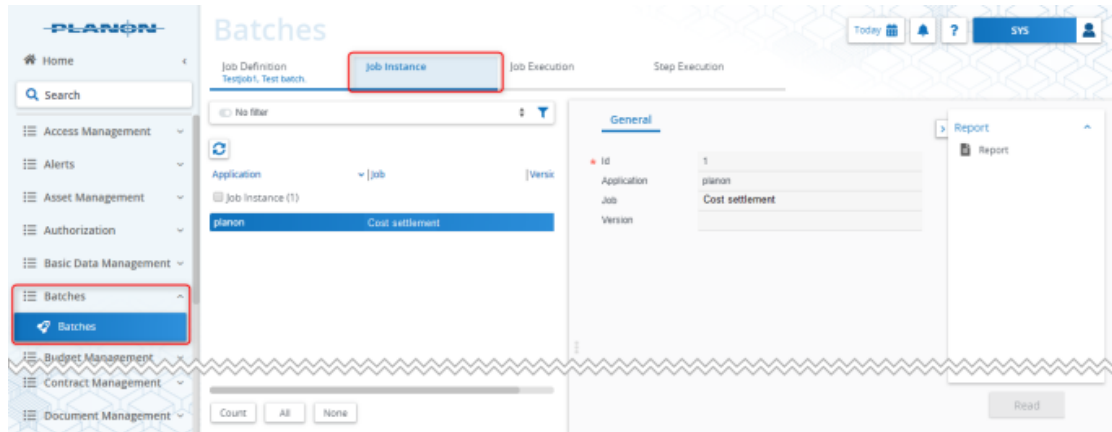
 For more information about configuring batch jobs, see the *javadoc* in your Planon installation folder:

```
..\related components\manual_installation_resources\sdk\QueryBuilderApi  
\com.planonsoftware.querybuilderapi<version>-javadoc.jar
```

## Job instance

When a batch job is started, a new job instance is created.

At the **Job instances** level, you can find basic information such as the application name and the batch job definition name.

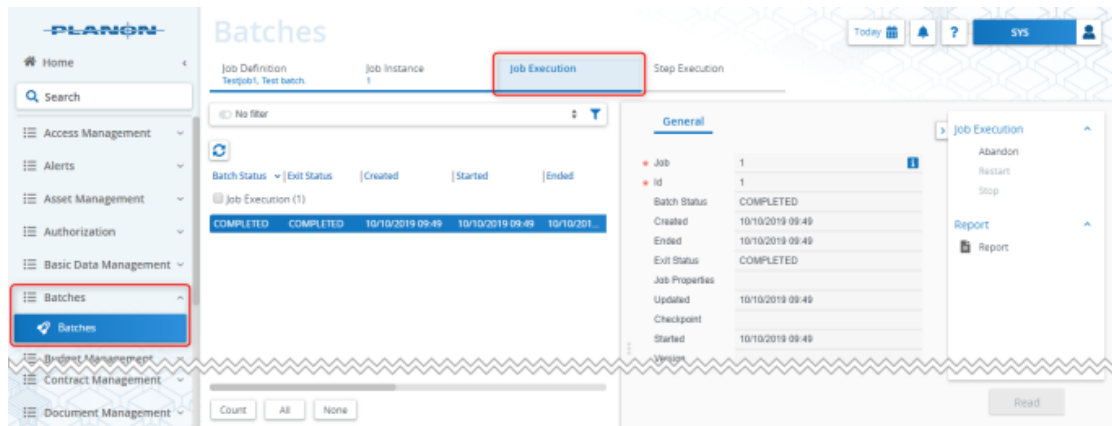


**i** A batch can only be started if there are no other batches in progress.

## Job execution

The batch job execution shows execution information of the job instance.

At the **Job execution** level you can find information such as when the job Instance was started, its current status, when it was ended, and if it was successful.



## Step execution

The **Step execution** level shows the execution information for the selected batch job.

The following image displays only one step in the batch job, which was successfully completed and has committed one item. In addition, it shows how many items are read, processed/skipped and/or written.



The screenshot displays the PLANON Batches Step Execution view. The sidebar on the left has 'Batches' selected. The main area shows a table with one row: 'Step 1' with a status of 'COMPLETED'. The 'General' details panel on the right shows the following information:

Job Execution	1
ID	1
Batch Status	COMPLETED
Committed	1
Ended	10/10/2019 09:40
Exception	
Exit Status	COMPLETED
Filter Count	0
User Data	
Process Step Count	0
Read count	0
Write Skip Count	0

If a step fails, this view will show the exception information and the cause of the failure.

# Index

## **B**

Batch processing 5

## **I**

Introduction 5

## **J**

Job definition 7

Job execution 8

Job instance 7

## **P**

Process overview 5

## **R**

Requirements 6

## **S**

Step execution 8

## **W**

Working with... 7