PI CONNECT Simatic Batch

User Guide

V3.1

Siemens AG, RC-DE I CS MTE MHM

Produced in Germany.

Siemens AG, RC-DE I CS MTE MHM

Table of Contents

1. Preface 5
   - 1.1 Audience 5
   - 1.2 Relationship between OSIsoft and Siemens software 5
   - 1.3 Organization 5
   - 1.4 Applicable other documents 6
   - 1.5 Submitting comments 6
   - 1.6 Copyrights and Trademark Notices 6

2. Function of the PI CONNECT SIMATIC Batch Interface 7
   - 2.1 Software overview 7
   - 2.2 Data Flow 8
   - 2.3 PI Connect Simatic Batch Service 9
   - 2.3.1 The Data Collector functionality (DC) 9
   - 2.3.2 The Data Manager functionality (DM) 10
   - 2.4 The Data Access API 10
   - 2.5 Automatic and manual historical data recovery 11
   - 2.6 Batch Dump Tool 11
   - 2.6.1 Using the tool 11
   - 2.6.2 Support for evaluation 13
   - 2.6.3 Support for manual historical data recovery 13

3. Configuration 14
   - 3.1 The Configuration application 14
   - 3.2 Helpful functions of the Configuration Application 14
   - 3.3 Instance configuration 16
   - 3.3.1 Misc parameters 17
   - 3.3.2 PI Data parameters 17
   - 3.3.3 SIMATIC Batch API parameters 18
   - 3.3.4 Trace parameters 19
   - 3.3.5 XML batch file handling parameters 20

4. Data description 21
   - 4.1 Supported Simatic Batch Level design 21
   - 4.2 The batch event structure 22
   - 4.2.1 Basic parameters 22
   - 4.2.2 Batch Structure Parameters 23
   - 4.2.3 Equipment structure parameters 24
   - 4.2.4 Event definition parameters 25
   - 4.3 Description of the batch event types 25
   - 4.3.1 Procedure start and end events 26
   - 4.3.2 Recipe header events 26
   - 4.3.3 State change events 27
   - 4.3.4 Process parameter value notification events 28
   - 4.4 Extended Batch event structure 29
   - 4.4.1 Event definition parameters (extended batch data) 29
   - 4.5 Description of the extended batch event types 30
   - 4.5.1 Operator comments: 30

4.5.2 Actions 31
4.5.3 Change of parameters 32
4.6 Restrictions and limitations 32
4.7 Example of the OSISoft interface PISISBatch.ini file 34

5. User Guide Revision History 35

1. Preface

1.1 Audience

The PI CONNECT Simatic Batch interface software was designed to be an open system. This guide is addressed to administrators and software engineers who are interested in function of the interface in detail, how to configure and what's about maintenance. For hardware/software prerequisites and installation refer the "PI CONNECT Simatic Batch- Installation Guide" document.

1.2 Relationship between OSIsoft and Siemens software

The interface software is split in two parts. The OSIsoft interface "PI Interface for Siemens SIMATIC Batch" and the Siemens interface "PI CONNECT Simatic Batch"

You need both interfaces to transfer data from SIMATIC Batch to OSIsoft PI batch. This document only describes the Siemens interface. The OSIsoft interface provides an own documentation.

The current release of the Siemens interface is designed and tested with the version 3.0.14.387 of the OSIsoft interface.

The OSIsoft interface normally is backward compatible. We recommend to ask OSIsoft for actual patches and releases.

1.3 Organization

Chapter 1	This chapter contains a description of the software structure and the mode of operation.
Chapter 2	This chapter contains a description of the software structure and the mode of operation.
Chapter 3	This chapter provides information required to configure the software.
Chapter 4	This chapter contains information about the structure and content of the Batch events from the Simatic Batch server. You need this information to build up the configuration of the OSIsoft interface. It also contains **Important Remarks**
Chapter 5	Tracing concept and overview of all important error massages
Chapter 6	Document history

1.4 Applicable other documents

All technical details and descriptions contained in our offer are valid and precede this document in case of differences e.g. in respect to released versions of the interface and the needed other software components (e.g. Operating system, Simatic PCS7, Simatic Batch, PI Simatic Batch API, Simatic Batch Interface to PI System), redundancy, supported character sets, support of flat recipes and others.

1.5 Submitting comments

Please feel free to give us your opinion of the usability of this manual, to suggest specific improvements, and to report errors. Mail your comments to:

Address	Siemens AG
	RC-DE I CS MTE MHM
	Dynamostr. 4
	D - 68165 Mannheim
	Germany
e-mail	info.rhm.industry@siemens.com

1.6 Copyrights and Trademark Notices

2. Function of the PI CONNECT SIMATIC Batch Interface

2.1 Software overview

The Siemens component of the interface consists of three functional units:

- Data Collector
- Data Manager

} PI Connect Simatic Batch Service

- Data Access API

The Data Collector connects the Simatic Batch server by using the Simatic Batch API.

The Data Manager receives the raw data from the Data Collector and transforms it into an event data format.

Since version 3 of the interface the Data Collector and Data Manager is combined in the PI Connect Simatic Batch Service.

The OSIsoft part of the Interface has access to the event data via the Data Access API.

The Data Access API is not an own process. The OSIsoft interface uses it to access to the Batch Data.

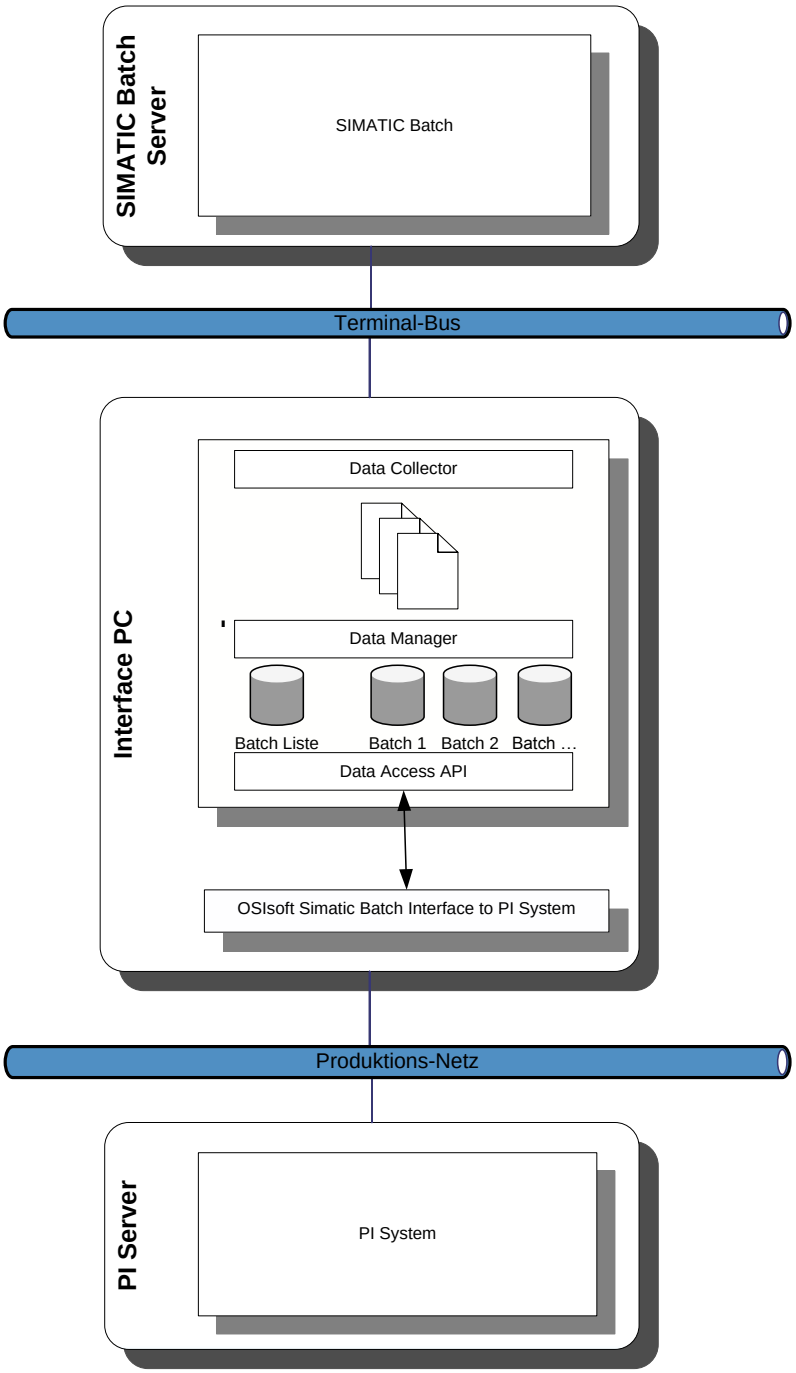
The following software must be installed and licensed on each Interface PC:

- PI System Develop Kit (SDK) [1]
- OSIsoft Interface “PI Interface for Siemens Simatic Batch” [1]
- Simatic Batch Client (no license needed at the Interface PC) [1]
- Siemens Interface “PI CONNECT Simatic Batch”
- SQL Server 2005 Express

[1] Not included into the PI Connect Simatic Batch interface delivery

2.2 Data Flow

This figure shows the dataflow from Simatic Batch to the PI system. Not shown are the communication interfaces to Simatic Batch (Simatic Batch Client within Simatic Batch API) and PI SDK (includes PI API and AF Client). The PI CONNECT SIMATIC Batch interface software part is shown at the yellow box.



2.3 PI Connect Simatic Batch Service

The PI Connect Simatic Batch Service (short SERVICE) runs as a service. About the detailed settings the service like account, startup mode and recovery please refer the document “PI CONNECT Simatic Batch- Installation Guide”.

The Service includes the Data Collection and Data Management functionality.

2.3.1 The Data Collector functionality (DC)

The Data Collector functionality (short DC) uses the Simatic Batch API to access Data from a Simatic Batch. The DC’s task is to receive all important batch data from Simatic Batch and store this data in raw format. This is the captured data:

- SIS events of Simatic Batch
- Material lists to provide code and name of used materials
- Data Type list to translate enumeration type IDs to text strings
- Object Data of a Batch to provide the Extended Batch Data

The DC needs a local installed Simatic Batch API (installed by Simatic Batch Client Setup). The local Simatic Batch API covers network connection to stand alone Simatic Batch or redundant Simatic Batch servers. The DC checks the connection in a defined cycle. The DC is normally configured to exit the service if the check fails or no data is arriving within a defined time range (see parameter “Time out of batch events” at chapter *3.3.3 SIMATIC Batch API parameters*). Therefore the service is configured to restart by service control manager (see also the document “PI CONNECT Simatic Batch- Installation Guide”.

After each connect or reconnect to the Simatic Batch the DC performs an archive recovery starting at the event ID of the last captured date. Thereafter it switches automatically to real time data processing. In case of the first start the DC does not perform a recovery. If you want to do this, please refer to chapter “2.5

Automatic and manual historical data recover".

To read the extended data of a batch (actions with signatures or operator comments) you can define up to 5 triggers after the end of a batch. By each trigger the complete extended data **available at this moment** will be read for this batch.

All extended data which are added in Simatic Batch after the last trigger will be ignored!

Please note: You need a license including the option "Comments" and/or "Actions & Signatures" to get extended batch data.

2.3.2 The Data Manager functionality (DM)

The Data Managers (short DM) function is to process all data from the DC, transform it to batch events and store these to a database. The structure of the events is adapted for the OSIsoft part of the interface and is independent from the Simatic Batch versions and project settings.

Another important functionality of the DM is the database management. The structured event data are not stored endless. The DM deletes all batches where end time is older than a configured time span. After deletion the batches are no longer available for the OSIsoft Interface. If you want to recover these batches again by the OSIsoft interface please see chapter*2.5*

Automatic and manual historical data recover.

An access to the DMs database is not allowed and can have inconsistent and destroyed data as a consequence!

2.4 The Data Access API

The OSIsoft Interface uses the Data Access API (short API) to query the Batch and extended Batch data from the DMs database. The Data Access API runs in the context of the OSIsoft interface process and has neither a user interface nor a configuration file.

The data format and mode of query was agreed by OSIsoft and Siemens. The Data Access Interface was designed as an API, which is a DLL, implemented by Siemens and used by the OSI PI-Batch interface.

The API is designed for exclusive use of OSIsoft interface. Any other use is not allowed and can have inconsistent and destroyed data as a consequence!

2.5 Automatic and manual historical data recovery

After restart or reconnect to Simatic Batch the DC normally recovers beginning at the last captured event ID. This ID is stored in the file “EventID_<Instance>.dat” located in the program data folder of the interface accessible in two ways:

- **%ProgramData%\Siemens\piconnectsimaticbatch**
- **Link at the PI CONNECT SIMATIC Batch configuration dialog.**

To perform a manually history recover follow the next steps:

- Stop the PI CONNECT Simatic Batch service of the instance where you want to do a recovery.
- Open the “EventID<Instance>.dat file using a text editor like notepad.
- Set the Event ID of the first event you want to capture from Simatic Batch. In case you want to recover all available data from SIMATIC BATCH set the ID to zero. Save the file.
  See chapter *2.6.3 Support for manual historical data recover”* to find out the Event ID.
- Open the Configuration Application *(*see chapter *3.1 The Configuration application*). Set the parameter “Minimum batch data buffering“ (chapter *3.3.1 Misc parameters)* to a time range which contains all recovered batch data. If the time range is set to small, the recovered batches will be deleted by the Data Manager immediately!
- Restart the PI CONNECT Simatic Batch service of the instance.
- Reconfigure and restart the OSIsoft interface to recover the data. Please read the OSIsoft manual for details.

Please keep in mind: depending on the amount of data to recover the Siemens and the OSIsoft interface software need CPU performance and time!

2.6 Batch Dump Tool

The setup also installs a helpful tool called “Batch Dump”. Using this tool you can dump all data of a list of batches from a SIMATIC Batch server to files.

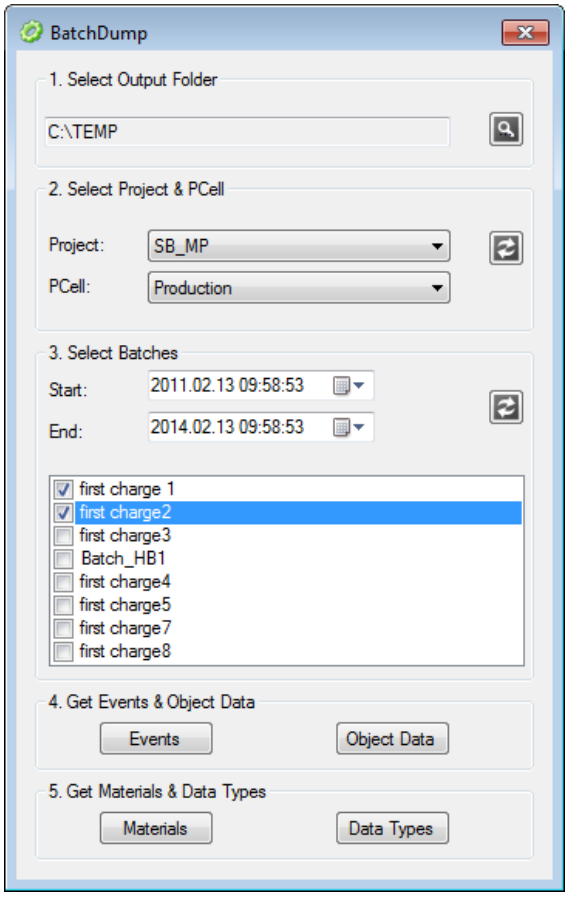
This tool is needed in case of checking compatibility of data, clarification of unexpected data transmission or to find the start Event ID for manual history recover:

2.6.1 Using the tool

You can start the Batch Dump tool at:

Start -> All Programs -> PI CONNECT SIMATIC Batch V3 -> Batch Dump

This screenshot shows the surface of the tool:



1. Select an output folder. Take care to have write access to this folder!

2. Select a Project and PCell by using the  to query the available items and select them on the drop down boxes

3. Select a time range and query the list of batches by pushing  button. Select the batches.

4. Press both buttons to get all SIS Events and Object Data of the selected batches.

5. Press both buttons to get additional information from the batch server

Thereafter the output folder contains <u>for each batch</u> an object data file:

<*BatchID*>_OBJECTDATA _<*timestamp*>.xml

and a batch event data file:

H_<*timestamp*>_576_0000000000_<*BatchID*>_<H or F>.xml.

The folder contains also a material list file (materials.xml) and a data type file (DATATYPES.xml):

2.6.2 Support for evaluation

In some cases we recommend an evaluation of the SIAMTIC Batch data before project start or before commissioning. In such case please install the software as described at the document “PI CONNECT Simatic Batch- Installation Guide”. You do not need to configure the software but the SIMATIC Batch API must have a connection to the Batch server. If you do not want to setup a new batch client station you can use an existing. After evaluation you can uninstall the software. We recommend making a system image or snapshot before installing the software.

If you send us evaluation data please select enough batches to have all kind of batch data in the dumps.

Best way is to select an empty output folder to dump all data, zip (compress) the folder and send the zip file by e-mail.

2.6.3 Support for manual historical data recovery

To start a manual historical data recovery you need a start BatchID. A start at the middle of a batch makes no sense. So you can use the Batch Dump tool to select the first batch you want to start from.

Get the event data of this batch and open the xml file by notepad. The next screenshot shows the EventID of the batches first event. Use this ID decremented by one as start BatchID.

3. Configuration

3.1 The Configuration application

The PI CONNECT SIMATIC Batch Interface provides a configuration application (short CONFIG-App). The description of creating a new instance, register it as a service, initialize the database and the license handling is described in the “PI CONNECT Simatic Batch- Installation Guide” document.

The User manual shows how to set the configuration parameters of an instance and some more helpful functions of the CONFIG-App.

You must have **administrator privileges** to start the CONFIG-App. You can find the CONFIG-App at the start menu:

Start -> All Programs -> PI CONNECT SIMATIC Batch V3 -> Configuration

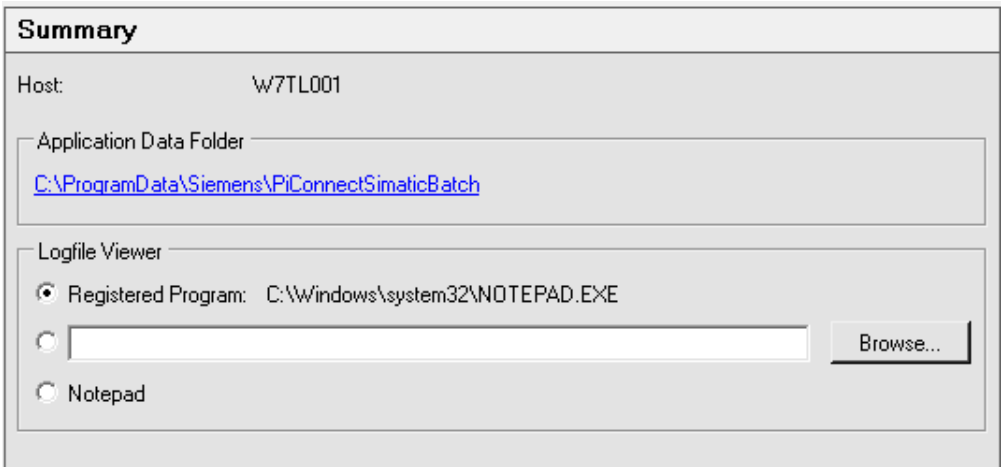
3.2 Helpful functions of the Configuration Application

If you select the root node (=host name) at the tree view, the CONFIG-App provides at the right side some helpful functions:

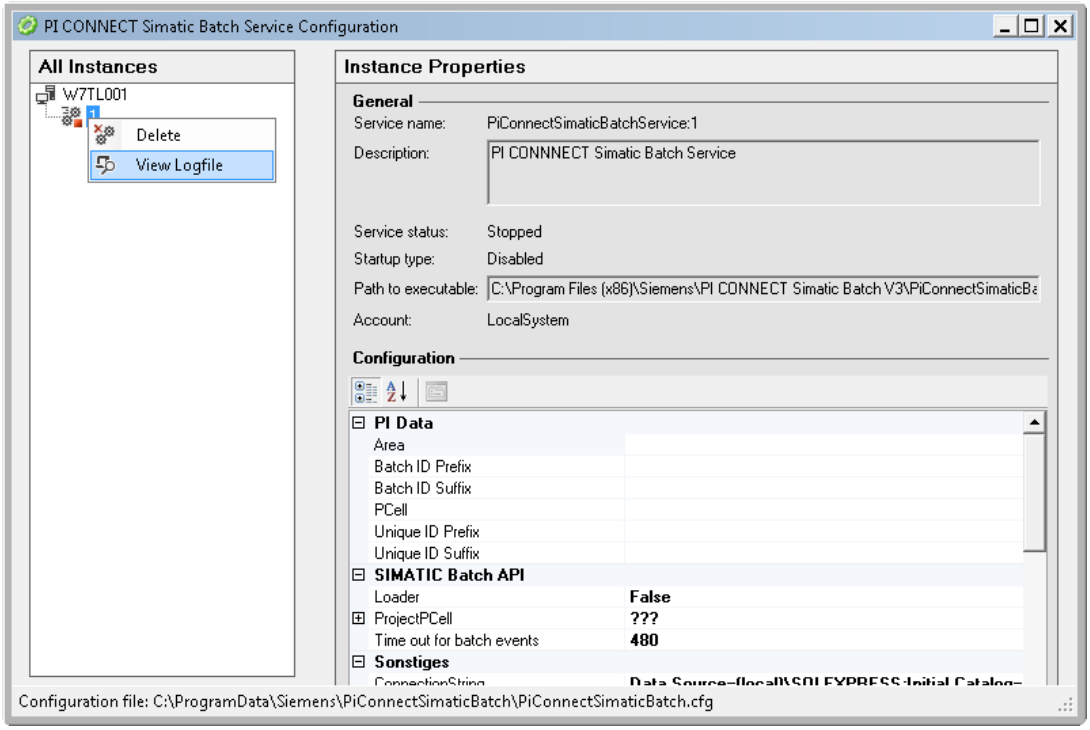
- **Shortcut to the Program Data folder**
  In this folder the interface stores the log files of the instances and the files containing the last captured event ID (used for automatically recover at startup).

- **Registration of a Logfile viewer**
  If you want to use an other program than notepad to view the log files, you can register your own application here.

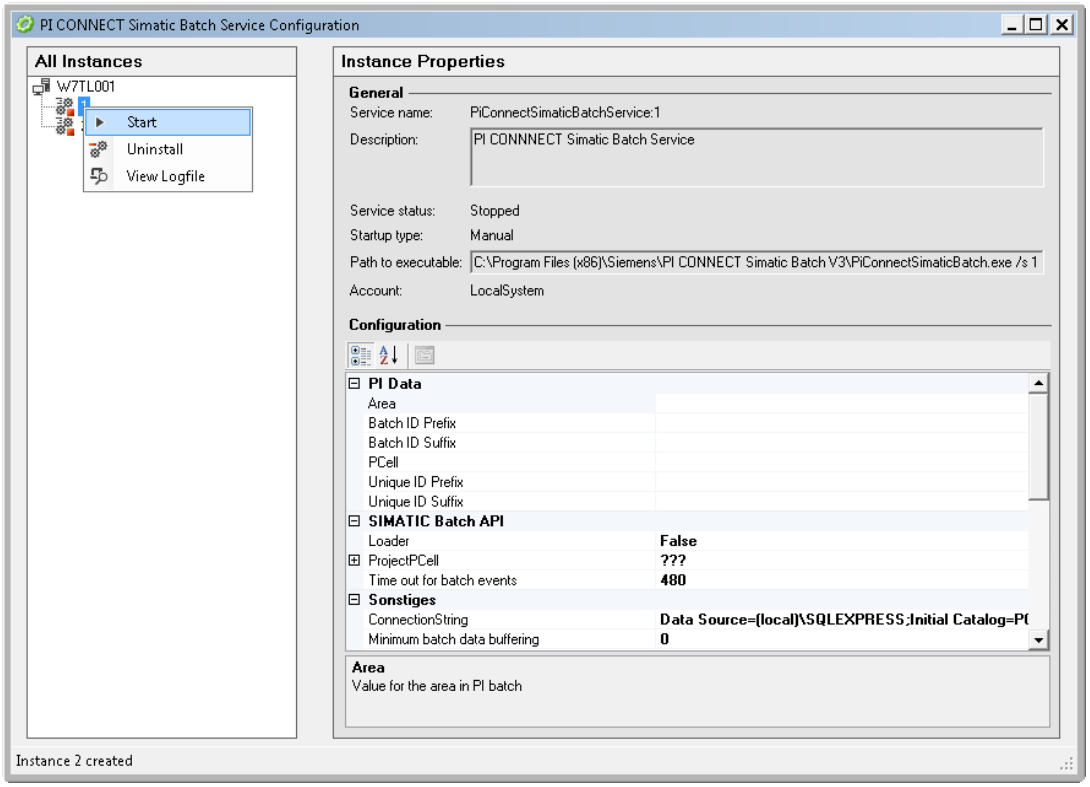
The next screenshot shows the functions on the right side of the CONFIG-App:



If you select an instance and open the context menu by right-mouse-click you can open the logfile of this service instance. At the right top side you can also see some general service information:

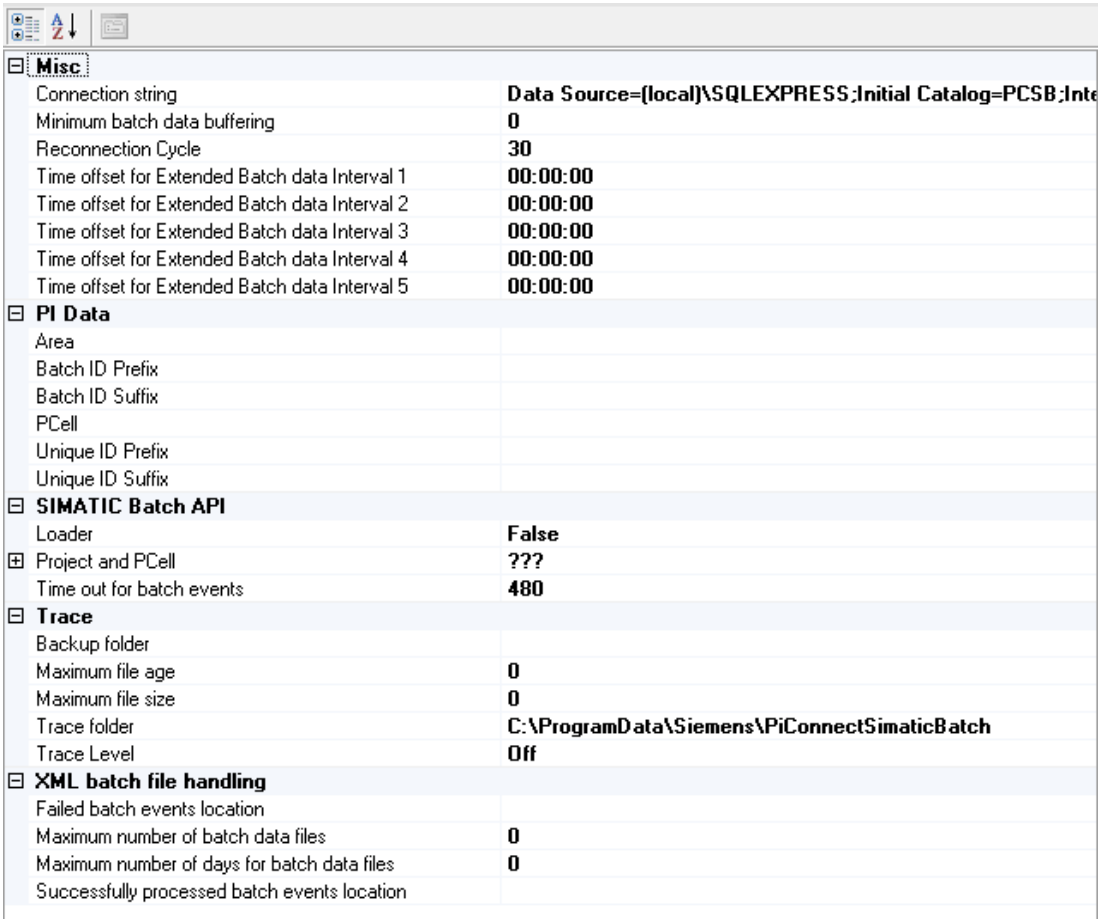


You also have the functions to start or stop the service of an instance by this context menu:



3.3 Instance configuration

If you select an instance at the tree view the CONFIG-App shows the configuration parameters of this instance on the right side.



Misc	
Connection string	**Data Source=(local)\SQLEXPRESS;Initial Catalog=PCSB;Inte**
Minimum batch data buffering	**0**
Reconnection Cycle	**30**
Time offset for Extended Batch data Interval 1	**00:00:00**
Time offset for Extended Batch data Interval 2	**00:00:00**
Time offset for Extended Batch data Interval 3	**00:00:00**
Time offset for Extended Batch data Interval 4	**00:00:00**
Time offset for Extended Batch data Interval 5	**00:00:00**
PI Data	
Area	
Batch ID Prefix	
Batch ID Suffix	
PCell	
Unique ID Prefix	
Unique ID Suffix	
SIMATIC Batch API	
Loader	**False**
Project and PCell	**???**
Time out for batch events	**480**
Trace	
Backup folder	
Maximum file age	**0**
Maximum file size	**0**
Trace folder	**C:\ProgramData\Siemens\PiConnectSimaticBatch**
Trace Level	**Off**
XML batch file handling	
Failed batch events location	
Maximum number of batch data files	**0**
Maximum number of days for batch data files	**0**
Successfully processed batch events location	

The default view shows the parameters functionally grouped by Misc, PI Data, SIMATIC Batch API, Trace and XML batch file handling. The following tables describe the parameters of each group.

3.3.1 Misc parameters

Parameter	Data type	Description	Default value
Connection string	String	Connection to the SQL Server database. Each instance have to use an own database. To set the value open the Connection Properties dialog. The “Log on to the server” settings are only used by this dialog (Test Connection Button and select a Database). The interface runtime uses an own account.	<empty>
Minimum batch data buffering	Integer	Numbers of days into the past, the batches are kept in the PI Connect Simatic Batch database. Set zero for endless storage (not recommended)	0
Reconnection Cycle	Integer	Seconds to wait after an unsuccessfully connect to SIMATIC Batch	30
Time offset for Extend Batch data Interval 1..5	Time	Time spans to wait after batch end or previous read of extended batch data. If you do not need all 5 times, set the further ones to zero. Format of the value: DAYS**.**HH**:**MM**:**SS	00:00:00

3.3.2 PI Data parameters

The parameters of this group complement the event data to OSIsoft PI Batch by constant values or allow making IDs unique. The interface does not use the Production Cell (PCell) name of Simatic Batch because a single instance is able to connect to one PCell only. Thus it is more flexible to have a SIMATIC Batch independent value for the PCell in the configuration.

Parameter	Data type	Description	Default value
Area	String	Value for the area in PI Batch. Set a unique value of Area/PCell for each instance of the interface.	<empty>
Batch ID Prefix	String	Prefix of the batch id.	<empty>
Batch ID Suffix	String	Suffix of the batch id.	<empty>
PCell	String	Value for the process cell in PI Batch. Set a unique value of Area/PCell for each instance of the interface.	<empty>
Unique ID Prefix	String	Prefix of the unique id.	<empty>
Unique ID Suffix	String	Suffix of the unique id	<empty>

Relation between OSIsoft PI Batch and Siemens SIMATIC Batch terms:

The PI term “Unique ID” means the “Batch ID” of SIMATIC Batch

The PI term “Batch ID” means the “Batch name” of SIMATIC Batch

In the following this document uses the OSIsoft PI Batch terms!

Batch/Unique ID Pre-/Suffix

Normally a single SIMATIC Batch server creates Unique IDs by incrementing the ID at each created batch. But in case of setting up a new SIMATIC Batch server, or restoring an image of a Batch Data Base this ID may step back and produce ambiguous IDs.

The consequences of this are overwritten and destroyed data in the OSIsoft PI Batch archive!

If you are not able to set the ID at Simatic Batch to a good (unique) value, you have to set prefixes and/or suffixes to get unique IDs.

Another case to get ambiguous values is a multi instance environment.

In such cases you have to set prefixes and/or suffixes at least for Unique ID. It is recommended to use the same prefixes and suffixes for Unique ID and Batch ID.

3.3.3 SIMATIC Batch API parameters

This group contains the connection parameter to SIMATIC Batch.

Parameter	Data type	Description	Default value
Loader	bool	API connection mode. Do not change without advice by support people.	False
Project and PCell	String	Select a connection to a SIMATIC Batch Server PCell using the “…” button in the value box. Please note: You must have a valid configuration for the Batch API and you have to be able to connect to the destination SIMATIC Batch Server.	Empty
Time out of batch events	Integer	Time in minutes an event from SIMATIC Batch is expected. After this time without an event the Service of the instance exits and will be restarted by the service control manager of Windows.	480

3.3.4 Trace parameters

This group contains the tracing control parameters.

Parameter	Data type	Description	Default value
Backup folder	String	Folder of old trace files. If value is empty the Trace folder is used.	Empty
Maximum file age	Integer	The number of days the trace files to be kept in the backup folder. Set to zero no traces will be deleted.	0
Maximum file size	Integer	Maximum file size in Kbytes of a trace file. Set to zero for maximum size of the file system	10000
Trace folder	String	Folder name all tracing information is directed to.	Program Data folder
Trace level	Enum	Controls the amount of trace data are written to the file. Highest level 'Verbose' is not allowed in productive environments.	Info

3.3.5 XML batch file handling parameters

After the Interface has processed a SIMATIC Batch event, it will be stored into a xml file, in case of successful processing located in the “Successfully processed batch events location” otherwise in the “Failed Successfully processed events location”. It is recommended to set at least a number of files limit.

Parameter	Data type	Description	Default value
Failed batch events location	String	Folder to store all failed processed SIMATIC Batch events. The value of this parameter should be left empty if you don’t want to store processed SIMATIC Batch events.	empty
Maximum number of batch data files	Integer	Number of files to be kept in the failed or successful batch events location. If set to 0 files will never be deleted by this mechanism (not recommended).	10000
Maximum number of days for batch data files	Integer	Number of days the files to be kept in the failed or successful batch events location. If set to 0 files will never be deleted by this mechanism	0
Successfully processed batch events location	String	Folder to store all successfully processed SIMATIC Batch events. The value of this parameter should be left empty if you don’t want to store processed SIMATIC Batch events.	empty

4. Data description

The PI CONNECT SIMATIC Batch Interface provides the data for the OSIsoft Interface. The OSIsoft Interface polls structured event data by the Data Access API from the Siemens interface.

This chapter describes the data structure and contents of all possible data events. **You need this information to build up the "Initialization file PISISBatch.ini" of the OSIsoft interface.** We provide a very simple example of the "Initialization file PISISBatch.ini" at the end of this chapter.

The Batch Data are UTF-16 coded by SIMATIC Batch and transferred without any conversion to the OSI PI-Batch Interface. Thereby all special characters are handed over. However the PI SDK supports 8 Bit ASCII. Thus the OSIsoft PI Batch Interface converts the data. For details please refer to the Interface description of the OISsoft Interface.

The solution (Siemens and OSIsoft part together) supports only German and English language and country settings. These settings have to be consistent in the complete system (Simatic Batch, Interface PC and PI Client).

The interface provides **batch event data** and optionally **extended batch event data**.

4.1 Supported Simatic Batch Level design

Simatic Batch normally organizes batches in a **hierarchical structure** with 4 levels:

- batch level
- unit procedure level
- operation level
- phase level

Some SIMATIC Batch projects have phases running direct under a unit. In this case the interface generates a pseudo operation and puts the phase information into this. The name of such operation is identical to the unit procedure name.

SIMATIC Batch also has a **flat structure** option with only 2 levels:

- Batch level
- Phase level

The Interface always creates 4 level batch events for the PI system. In case of flat structure the interface uses the “Unit” parameter of each SIMATIC Batch phase to create the unit procedure level. The name of the unit procedure and the unit (equipment) is identical. The interface also creates a pseudo operation level with the same name as used for the unit.

Please note: The handling of flat or hierarchical batch data is both a license option. If you e.g. don’t have a license for flat batches, such batch data will be ignored. A mixed license is possible.

4.2 The batch event structure

This chapter describes the structure and the meaning of all event types. Each parameter listed in the following can be used in the “Initialization file PISISBatch.ini” except the hidden parameters level and TimeStampUTC. These are for internal use of the OSIsoft interface only.

The parameters are divided in basic parameters, structure parameters, event definition parameters and equipment structure parameters. Here the complete list of all parameters.

4.2.1 Basic parameters

Each event provided by the PI Connect Simatic Batch interface contains the following header Information:

parameter	Meaning	optional / obligatory
*Level (*1)*	***1**: Batch* ***2**: Unit Procedure* ***3**: Operation* ***4**: Phase*	*obligatory**, hidden***
*TimeStampUTC (*1)*	*Timestamp of the event in seconds since 01.01.1970. Floating point value containing sub seconds.*	*obligatory**, hidden***
Time	GMT Timestamp of the event in ‘yyyy-MM-dd hh:mm:ss.000’ format. Formatted by the OSI Interface.	obligatory

*1: hidden, user can’t access by OSI Interface Template

4.2.2 Batch Structure Parameters

The structure parameters describe all what you need to assign the event to a specific Batch element (batch, unit procedure, operation or phase) including multiple activations of the element. The Unit-, Container- and Terminal - IDs can be used to get unique names of the batch elements in the OSI PI Server and to refer the recipe structure of SIMATIC Batch. The following table shows the batch structure parameters of an event. Naturally depending on the level (Batch, Unit procedure, .) not all parameters are set.

parameter	Meaning	optional / obligatory
UniqueID	Unique ID of the PI Batch (= Simatic Batch "Batch ID", must be unique at any time of the lifecycle)	obligatory
BatchID	Batch ID of the PI Batch (=Simatic Batch "Batch Name")	obligatory
Procedure	Name and version of the master recipe	obligatory
UnitProcedure	Name of the unit procedure	Level 2,3,4 : obligatory Level 1 : empty
UnitActivation	Number of the unit procedure activation	Level 2,3,4 : obligatory Level 1 : empty
UnitContID	Identifies uniquely a unit procedure within an batch	Level 2,3,4 : obligatory Level 1 : empty
UnitID	ID of the unit (equipment, no procedure)	Level 2,3,4 : obligatory Level 1 : empty
Operation	Name of the operation	Level 3,4 : obligatory Level 1,2 : empty
OperationActivation	Number of the operation activation.	Level 3,4 : obligatory Level 1,2 : empty
OperationContID	Identifies uniquely a phase within a unit procedure.	Level 3,4 : obligatory Level 1,2 : empty
Phase	Name of the Phase	Level 4 : obligatory Level 1,2,3 : empty
PhaseActivation	Number of the phase activation.	Level 4 : obligatory Level 1,2,3 : empty
PhaseContID	Reference to the ContID of the parent (operation or unit)	Level 4 : obligatory Level 1,2,3 : empty
PhaseTermID	Identifies uniquely a phase within an operation.	Level 4 : obligatory Level 1,2,3 : empty

4.2.3 Equipment structure parameters

To support the equipment structure in the OSI PI server the interface also provides structure information. This means the name of the units, operation modules and phase modules. In SIMATIC Batch an operation module and phase module name always equals to the operation and phase name. The equipment structure also contains the PCell and Area values (see chapter *3.3.2 PI Data parameters*).

parameter	Meaning	optional / obligatory
Area	first level of equipment structure in PI Batch	configuration
ProcessCell	second level of equipment structure in PI Batch	configuration
Unit	name of the unit module for PI Batch equipment structure (= Simatic Batch unit name)	Level 2,3,4 : obligatory Level 1 : empty
Operation Module	name of the opration module for PI Batch equipment structure (=Simatic Batch operation name)	Level 3,4 : obligatory Level 1,2 : empty
PhaseModule	name of the phase module for PI Batch equipment structure (=Simatic Batch phase name)	Level 4 : obligatory Level 1,2,3 : empty
ExtValue	for future use	always empty

4.2.4 Event definition parameters

The last section of each event is the “event definition” section. It includes all what happened at the batch element. The usage and meaning of the parameters in the 3 tables above is unique for each event type. The meaning of the parameters in the following table depends on the parameter event.

Parameter	Meaning	optional / obligatory
Category	**parameter type category:** • ProcessInput • ProcessOutput • ProcessParameter • ControlStrategy	Obligatory for parameter events • <Act_Val> • <SetPoint>
Descript	**Event <StateChange>:** • string value of the new state **Event <Act_Val> / <SetPoint>:** • parameter name **Event <Recipe Header>:** • recipe parameter name	obligatory for Event: • <Act_Val> • <SetPoint> • <Recipe Header> • <StateChange> all other empty
Event	**start / end events for all procedures(*1)** • <System> **batch / recipe header data:** • <Recipe Header> **state change for al procedures (*1):** • <StateChange> **parameter value event (all levels):** • <Act_Val> • < SetPoint>	Obligatory
Value	**Event <StateChange>:** • integer value of the new state **Event <Act_Val> / <SetPoint>:** • parameter value **Event <Recipe Header>:** • recipe parameter value **Event <System>:** • <Start> or <End>	Obligatory
Eu	engineering units: **Event <Act_Val> / <SetPoint>:** • Engineering units of the parameter. Not all parameters provides engineering units **Event <Recipe Header>:** • recipe parameter engineering units	Optional
MaterialName	name of the material	obligatory for Event: • <Act_Val> • <SetPoint> all other empty
MaterialID	ID of the material	obligatory for Event: • <Act_Val> • <SetPoint> all other empty
ParameterID		

4.3 Description of the batch event types

This chapter describes all type of events that can occur. Depending on your SIMATIC Batch project not all types of events may occur in all structure levels (batch, unit procedure, operation and phase). The following description explains the meaning of the event definition parameters. All other parameters do not depend on the event type.

The Interface provides 4 different event types:

- Procedure start and end events
- Recipe header events
- State change events
- parameter value notification events

The extended event types are described at an own chapter.

4.3.1 Procedure start and end events

These events are the first or last event of each element. An element can be a batch, unit procedure, operation or phase. To ensure that such an event will be the first or last of the procedure generated in PI Batch, the source timestamp from SIMATIC Batch will be decreased/increased for one millisecond by the interface software.

List of all used event definition parameters:

parameter	Meaning
Event	**System**
Value	**Start** or **End**

4.3.2 Recipe header events

For each procedure in SIMATIC Batch some header information is available. This Information depends on the SIMATIC Batch version, the project settings and the procedure level. The list below shows al possible events. If the SIMATIC Batch server does not deliver the information, the interface can't create such events.

The table below shows the available data for all four levels

Recipe Header Data	Batch	Unit Procedure	Operation	Phase
Formular_Name	**x**			
Formular_Version	**x**			
Master_Recipe_Name	**x**			
Master_Recipe_Version	**x**			
Start_Time_Planed	**x**			
End_Time_Planned	**x**			

Quantity (with eng. units if available)	**x**			
Product_Name	**x**			
Product_Code	**x**			
Order_Category_Name	**x**			
Order_Name	**x**			
Description	**x**			
Recipe_Structure (flat or hierarchical)	**x**			
Operator_Dialog_Type			**x**	**x**
Runtime	**x**	**x**	**x**	**x**

List of all used parameters:

Parameter	Meaning
Event	**Recipe Header**
Descript	Parameter name from list above
Value	Value of the parameter
Eu	engineering units, optional

4.3.3 State change events

Each level has state change events. The available states depend on the level.

Batch level states:	All other levels states:
02 PLANNED	01 EDITED
03 CANCELED	02 IDLE
04 RELEASE_PREPARED	03 STARTING
05 RELEASED	04 RUNNING
06 BLOCKED	05 READY
07 WAITING	06 COMPLETING
08 RUNNING	07 COMPLETED
09 PAUSED	08 CONTINIOUS
10 HELD	09 ABORTED
11 ABORTED	10 ABORTING
12 COMPLETED	11 STOPPED
13 STOPPED	12 STOPPING
14 RELEASE_CHECKING	13 HELD
15 RELEASE_INVALID	14 HOLDING
	15 RESUMING
	16 PAUSED
	17 PAUSING

Please note:

State events less than 07 WAITING (batch level) or 03 STARTING (other levels) are normally not processed by the interface. It is not possible to store such early events in OSIsoft PI Batch. Depending on your SIMATIC Batch version and project settings not all states will be provided by the SIMATIC Batch API.

Description of the state change event:

Parameter	Meaning
Descript	string value of the state (see listing above)
Event	**SateChange**
Value	Integer value of the new state (see listing above)

4.3.4 Process parameter value notification events

The Interface can process “Parameter Events” on all batch levels. Depending on the SIMATIC Batch configuration such events may appear on all levels or not. SIMATIC Batch knows four types of process parameters with different data properties:

Data property	Process Parameter	Process Input	Process Output	Control Strategy
Parameter name	x	x	x	x
Parameter ID	x	x	x	x
Actual value / set point	x	x	x	x
Engineering units	optional	optional	optional	
Material name		x	x	
Material code		x	x	

List of all used parameters for actual or set point value events:

Parameter	Meaning
Category	parameter type: • **ProcessInput (PI)** • **ProcessOutput (PO)** • **ProcessParameter (PP)** • **Control Strategy**
Descript	Simatic Batch parameter name
Event	**<Act_Val>** or **<SetPoint>**
Value	The value of the event
Eu	engineering units (optional)
MaterilName	name of the material (only PI and PO)

MaterialID	ID code of the material (only PI and PO)
ParameterID	Simatic Batch Parameter ID
ExtValue	Numeric index. Only in case the parameter is of data type enumeration

4.4 Extended Batch event structure

The extended batch data are a part of the batches object data. In the current release actions including a signature and operator comments are supported.

The object data are not transmitted online. The interface reads and provides such data 5 times after the end of the batch (see chapter *3.3.1 Misc parameters*). The main part of the event structure is equal in contend and meaning to the normal event data. You can see chapter about the parts:

- Basic parameters
- Batch Structure Parameters
- Equipment Structure Parameters

The content and structure of the event definition is different. Extended batch data have an own event definition structure described at the following.

4.4.1 Event definition parameters (extended batch data)

parameter	Meaning	optional / obligatory
ActionType	Type of extended batch data • **CREATED** • **PERFORMED** • **SIGNATURE** • **CHANGE** • **COMMENT**	Obligatory
ActionID	ID of the Action or ID of the operator comment	Obligatory
ActionName	Name of the action. Empty for ActionType COMMENT	Obligatory with the exception ActionType COMMENT
event	ActionType CHANGE: • **SetPoint**: Changes of parameter set point	Obligatory for ActionType CHANGE. All other empty
Descript	ActionType CHANGE: name of the parameter	Obligatory for ActionType CHANGE. All other empty
ParameterID	ActionType CHANGE: ID of the Parameter	Obligatory only for ActionType CHANGE

Value	ActionType CHANGE: the new value of the parameter	
Eu	ActionType CHANGE: engineering units if available	
OldValue	ActionType CHANGE: the previous value of the parameter	
UserID	Name of the user	Obligatory for CREATED, PERFORMED, SIGNATURE and COMMENT Actions. Empty for CHANGE Actions
LongUserID	Full name of the user	Obligatory for CREATED, PERFORMED, SIGNATURE and COMMENT Actions. Empty for CHANGE Actions
NodeName	Note who the action is created, performed or signed	Obligatory for CREATED, PERFORMED, SIGNATURE and COMMENT Actions. Empty for CHANGE Actions
Status	Actual status of the action. At ActionType CREATED and PERFORMED normally 'Closed'. At ActionType SIGNATURE 'SIGNED'. At ActionType CHANGES and COMMENT empty	Obligatory for CREATED, PERFORMED, SIGNATURE Actions. Empty for CHANGE and COMMENT Actions
Comment	In current version only set for ActionType COMMENT. Depending on the Simatic Batch version in future set also for ActionType SIGNATURE	Obligatory for ActionType COMMENT. All other empty.

4.5 Description of the extended batch event types

4.5.1 Operator comments:

The PI Connect SIMATIC Batch interface captures the operator comments of SIMATIC Batch. Each performed operator comment in SIMATIC Batch creates one single event for the OSIsoft interface.

Here the list of all used event definition parameters:

Parameter	Meaning
ActionType	**COMMENT**
ActionID	ID of the operator comment
UserID	name of the performing user
LongUserID	full name of the performing user
NodeName	name of the performing node
Comment	Comment text, **limited to 128 characters**

4.5.2 Actions

The PI Connect SIMATIC Batch interface captures all changes of batch parameters and all actions to the batch (hold, resume, …) which have a signature.

Actions without signature are not provided!

Actions occur on a specific batch element indentified by the batch path structure of the events. A batch element can have multiple actions.

An action can have multiple events. The Action ID allows assigning the action event to a specific action. In other words: the CREATED, PERFORMED SIGNATURE and CHANGE events of one action have the same unique ActionID.

For each action the PI Connect SIMATIC Batch interface creates a CREATE and a PERFORMED event within the following data:

Parameter	Meaning
ActionType	**CREATED** or **PERFORMED**
ActionID	ID of the action
ActionName	Name of the action
UserID	name of the performing user
LongUserID	full name of the performing user
NodeName	name of the performing node
Status	status of the action, normally ‘closed’ after batch end

For each transmitted action the interface creates also one or more SIGNATURE events (depending on the performed signatures at Simatic Batch) containing the following data:

Parameter	Meaning
ActionType	**SIGNATURE**
ActionID	ID of the action
ActionName	Name of the action
UserID	name of the signing user
LongUserID	full name of the signing user
NodeName	name of the performing node
Status	status of the signature, normally ‘SIGNED’ batch end

4.5.3 Change of parameters

If a user changes parameters in SIMATIC Batch and this action needs a signature, the PI CONNECT SIMATIC Batch interface creates a parameter change event of type SetPoint:

Parameter	Meaning
ActionType	**CHANGE**
ActionID	ID of the action
ActionName	Name of the action
Event	**SetPoint**
Descript	Name of the changed parameter
ParameterID	ID of the changed parameter
Value	new set value
EU	engineering units of the value if available
OldValue	previous value
LongUserID	full name of the signing user
NodeName	name of the performing node
Status	status of the signature, normally 'SIGNED' batch end

4.6 Restrictions and limitations

The very most of the important data for a processed batch is provided by the Data Access API to the OSIsoft interface. The SIMATIC BATCH Source events (SIS events) contain a lot of minor interesting data like internal IDs and others. However, only the described data is transmitted. It depends on the OSIsoft Interface configuration, which of this data will be stored in PI Batch.

The general structure of a batch is always transmitted completely with some little restrictions. In the following these restrictions are described.

Sub Structures

In flat recipes of SIMATIC Bach the structure element "Sub" may be used for grouping phases. Such a Sub can group phases running at different units. Subs can also contain Subs in any deep structure.

These two issues are not compatible to the structure aft OSIsoft PI Batch. So we don't transmit any Sub structure. We only ignore the Sub but we transmit all contained phases!

Monitors

SIMATIC Batch knows a control structure called Monitor. These can observer a part of a recipe and take actions on it. We do not support Monitors with the actual release.

All information about the monitor and all functions within it are not transmitted. But the result of the monitor's action like terminating a phase is of course transmitted by the interface.

Transitions, branches, loops and jumps

These structures normally have a condition which controls there function. We do not transmit any information about enter and leave of it and also no information about the condition. Such structures do not have an equivalent in OSIsoft PI Batch.

Command Step

Command steps are recipe elements which take action to one or more destination elements. The occurrence of the Command Step is not transmitted by the interface. Only the changes (e.g. aboard) on the destination element will be transmitted.

Object Data (extended events)

From the Batch object data we transmit only the above listed data. Especially user actions are only transmitted if they have a signature.

Some more data transmission is scheduled for the next release. Please feel free to request for if you miss any important data.

SIMATIC Batch AS Execution.

This new feature is not supported in the current release. Please contact us if you need this functionality.

4.7 Example of the OSISoft interface PISISBatch.ini file

The following box shows a very simple initialization file of the PISISBatch interface of OSISoft. Using the file the OSISoft interface creates unique names for all recipe levels and stores all events as raw data to two string tags. One tag to store normal event data the other tag to store extended event data. You can import the tag data of one batch to EXCEL, split them by the '|' character and so you get a complete overview of all available events of this batch.

```
Recipe[1].BatchID=@BATCH: [BatchID]([UniqueID])
Recipe[1].Name=Proc: [Procedure]
Recipe[2].BatchID=@RUP: [UnitProcedure]_[UnitContID]_[UnitACtivation]
Recipe[2].Name=Proc: [UnitProcedure]
Recipe[3].Name=@ROP: [Operation]_[OperationContID]_[OperationActivation]
Recipe[4].Name=@RPH: [Phase]_[PhaseTermID]_[PhaseContID]_[PhaseActivation]

Tag[1].Name = AllEvents
Tag[1].Trigger = [Event, value="System"]
Tag[1].Trigger = [Event, value="SetPoint"]
Tag[1].Trigger = [Event, value="Act_Val"]
Tag[1].Trigger = [Event, value="State Change"]
Tag[1].Trigger = [Event, value="Recipe Header"]
Tag[1].Value =
[TIMESTAMP]|[UniqueID]|[BatchID]|[Procedure]|[UnitProcedure]|[UNITACTIVATION
]|[UnitContID]|[UnitID]|[Operation]|[OPERATIONACTIVATION]|[OperationContID]|
[Phase]|[PHASEACTIVATION]|[PhaseContID]|[PhaseTermID]|[Category]|[Descript]|
[Event]|[Value]|[EU]|[Area]|[PROCESSCELL]|[Unit]|[OperationModule]|[PhaseMod
ule]|[MaterialName]|[MaterialID]|[ParameterID]|[ExtValue]

Tag[2].Name = AllExEvents
Tag[2].Trigger = [ACTIONTYPE, value="SIGNATURE"]
Tag[2].Trigger = [ACTIONTYPE, value="CREATED"  ]
Tag[2].Trigger = [ACTIONTYPE, value="PERFORMED"]
Tag[2].Trigger = [ACTIONTYPE, value="COMMENT"  ]
Tag[2].Trigger = [ACTIONTYPE, value="CHANGE"   ]
Tag[2].Value =
[TIMESTAMP]|[UniqueID]|[BatchID]|[Procedure]|[UnitProcedure]|[UNITACTIVATION
]|[UnitContID]|[UnitID]|[Operation]|[OPERATIONACTIVATION]|[OperationContID]|
[Phase]|[PHASEACTIVATION]|[PhaseContID]|[PhaseTermID]|[ACTIONTYPE]|[ACTIONID
]|[ACTIONNAME]|[Event]|[Descript]|[ParameterID]|[Value]|[EU]|[OLDVALUE]|[USE
RID]|[LONGUSERID]|[NODENAME]|[STATUS]|[COMMENT]|[Area]|[PROCESSCELL]
```

5. User Guide Revision History

Publication Date	Author	Manual Revision	Remarks
2009-12-07	Lutter	V1.0	Initial
2010-03-11	Hetterich	V1.01	Minor corrections and additions
2011-06-09	Hetterich	V1.02	Importance of unique BatchID
2012-12-03	Bäuerle	V1.03	Minor corrections
2013-02-13	Lutter	V2.2.0	batch event structure extensions.
2013-05-22	Bäuerle	V2.2a	Minor corrections
2013-10-14	Lutter	V3.0	Updating V3.0
2014-02-26	Lutter	V3.1	• service do not need administrative privileges • multiple instances • extended data