TECHNICAL UNIVERSITY OF CLUJ-NAPOCA

ACTA TECHNICA NAPOCENSIS

Series: Applied Mathematics, Mechanics, and Engineerin
Vol. 61, Issue II, June, 2018

SAP PLANT CONNECTIVITY

Dan BIZUBAC, Marcel Sabin POPA, Bernd Otto HÖRMANN

Abstract: *SAP Plant Connectivity (PCo) is a framework, set of services and management tools that enables the exchange of data between a SAP system and the industry-specific standard data sources of different manufacturers, such as process control systems, control devices, plant historian systems, SPC systems, TCP Sockets, and SAP Manufacturing Products (SAP ME, SAP MII, SAP EWM, SAP ODA and so on). With this application companies can receive tags and events from the connected source systems in production either automatically or upon request and forward them to the connected SAP systems. Furthermore, using PCo, manufactures can execute, receive, and process OPC UA method calls and thereby also map complex coordination tasks in the programmable logic controllers (PLC) environment. The research is carried out with the support of the Technical University of Cluj-Napoca.*

Key words: *Automation, Industrial Automation, Process, Lifecycle, Manufacturing*

1. INTRODUCTION

SAP Plant Connectivity is a Microsoft .Net-developed software component of SAP Manufacturing. PCo together with SAP Manufacturing Execution (ME) and SAP Manufacturing Integration and Intelligence (SAP MII) forms the comprehensive SAP Manufacturing Execution Suite, which gives companies the means to implement new dimensions of shop floor automation

The software tool named Management Console by SAP is an WinForms software application that configures the management of the PCo Source Agents, Destinations, Agent Instances and Notifications.

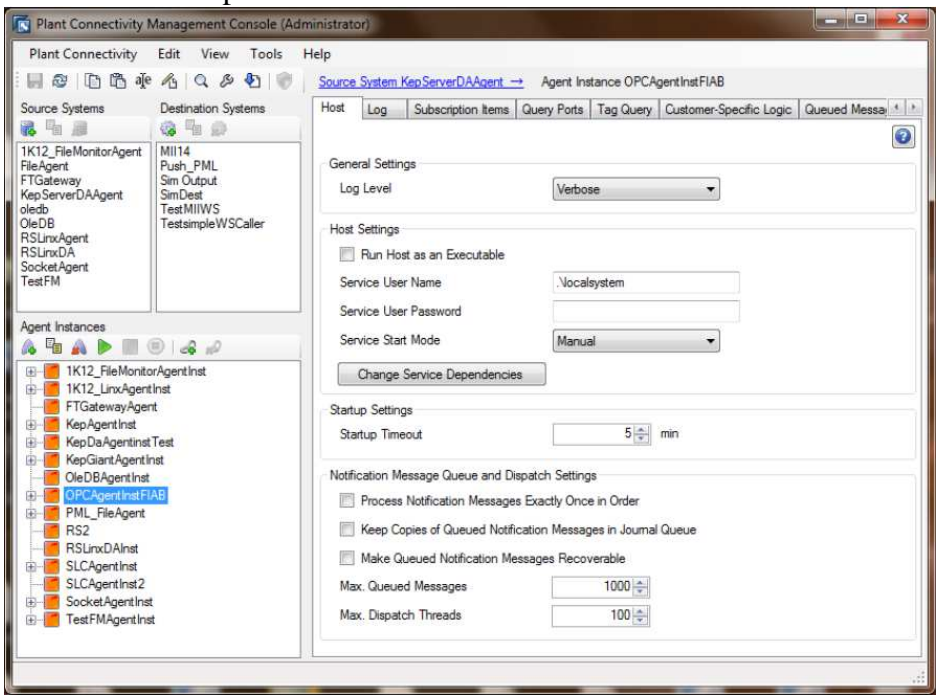

Fig1. Plant Connectivity Overview

1.1.Plant Connectivity Source Systems

The source system refers to the actual location where the data comes from and it needs to be retrieved. There is a need to create the connection from the PLC (Programmer Logic Controller) to the SAP Plant Connectivity tool.

The OPC Source System Agents are connecting to an OPC instance Server which will contain several interfaces in the Source System Agents. The OPC usually serves as a provider for raw data and notification interface to the actual controlling system without knowing the communication infrastructure and the actual drivers used. Some examples of OPC Systems may contain: Kepware KepServerEX, Matrikon OPC, Siemens Simatic.Net OPC etc.

There can be defined more than one source and more than one destination. Each agent is an instance of a particular source. Subscription elements can be browsed of a certain source in an agent if this agent is an instance of that particular source.

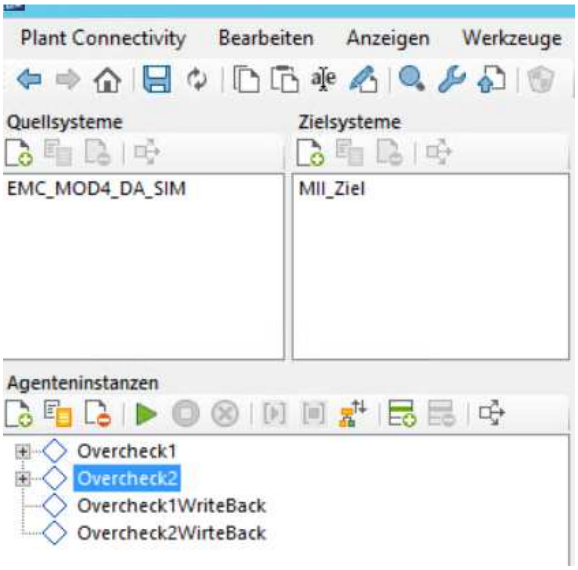

Fig. 2. Source Systems

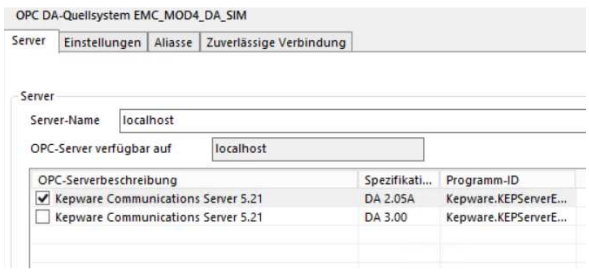

Fig. 3. Kepware Sources

A following example of an OPC Kepware data source system is configured (Fig. 3). If the OPC server is running on the same host as PCo the available OPC Servers should be available to select.

1.2.Agent and Notification

The Agents and Notification are the core of SAP PCo. The Agent is the connection to the OPC server. If an agent is started up by selecting an agent and pressing the “play” symbol the agent will start up and the white symbol in front of the agent turns green. This indicates that the agent is active and during this time any settings within the agent and the belonging notifications are not changeable.

The agent then listens to the items he has subscribed to (see subscription elements). The notification under an agent is doing the actual job. A notification is an instance of an agent. That means that a notification can only access tags the agent which is above it has subscribed to. In the notification you can define the destination system and the output parameters if the trigger condition is true the xml is sent to the destination.

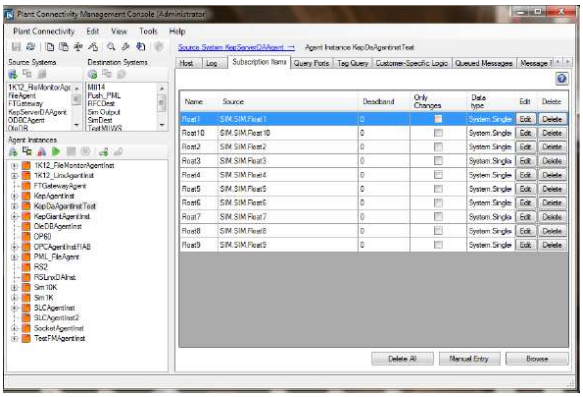

Fig. 4. Agents and Notifications

The agent contains on one side Notification and Subscription items on the other side.

1.3.Subscription Items

Under the tab subscription elements the elements from the OPC server can be defined and the agent is subscribing to these. There are several options as shown in the picture.

Elements can be added both manually as well as editing and deleting an existing one or even browsing for elements.

By browsing tags a direct access the OPC server and its data structure can be maintained. The required tags can be selected there and all what is needed is them to be added to the instance of the agent.

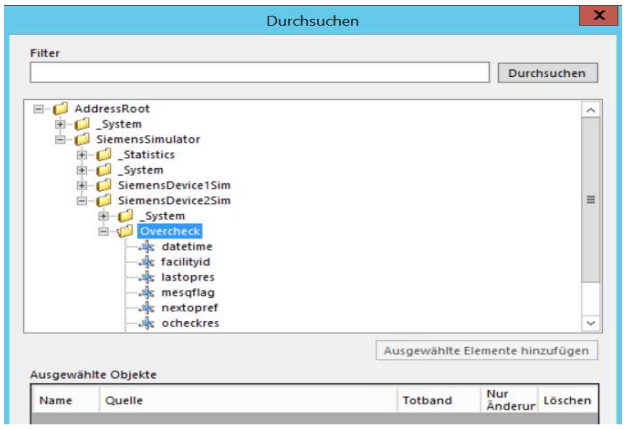

Fig. 5. Agents and Notifications

The Subscription Item list is similar to the one in the picture below as an example for the source path for instance "SiemensSimulator.SiemensDevice1Sim.Overcheck.datetime" is being selected.

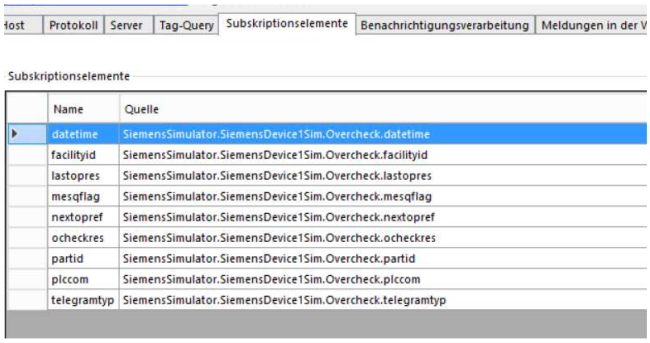

Fig. 6. Subscription List

The Notifications are entities that can detect easily occurring events in the system and will retrieve the data and send it further if the trigger expression is being met. The trigger notification

- Always: Notification Output is executed when any of the Agent Instance Subscription Tags changes value.
- OnTrue: Trigger expression changes from False to True after a Trigger Tag Value change
- WhileTrue: Trigger expression evaluates to True after a Trigger Tag Value change
- OnFalse: Trigger expression changes from True to False after a Trigger Tag Value change
- WhileFalse: Trigger expression evaluates to False after a Trigger Tag Value change

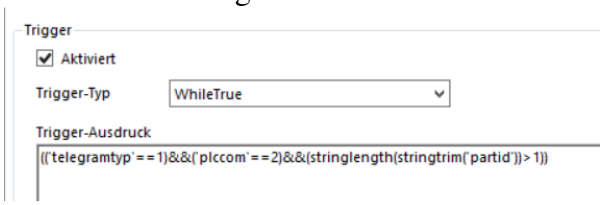

Fig. 7. Trigger Condition

1.4.Notification Output

The output of a notification defines which data is sent to the destination system. The data which is send can be either the tag information from the OPC server or it can also defined as a fix value which will not change.

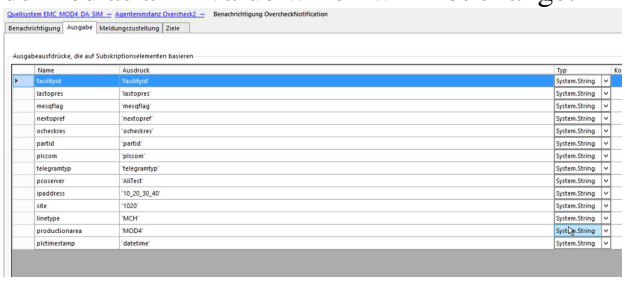

Fig. 8. Output Items

The configuration for the notification output can be done in different ways: a new output can be added, changed or deleted. By generating outputs, PCo will simply list all subscription elements of the agent as output.

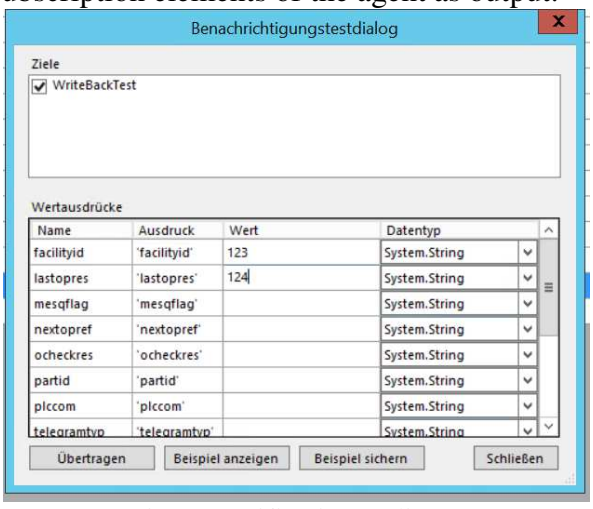

Fig. 9. Notification Delivery

1.5.Plant Connectivity Destination Systems

Usually the easiest way for the communication with the SAP Manufacturing Execution is through XML documents in the Manufacturing Intelligence and Integration Destination. All Notifications with a MII destination will send their XML output to this destination and the targeted transaction. In the destination server tab the server address can be defined for the targeted destination without "http" prefix. The port of the destination system and the version is required and the user who has access to the server. By clicking the "test connection" button a test can be performed to see whether PCo can connect successfully or not. If an error appears than there is a need to check the user and the server name in order to make sure you have the correct one.

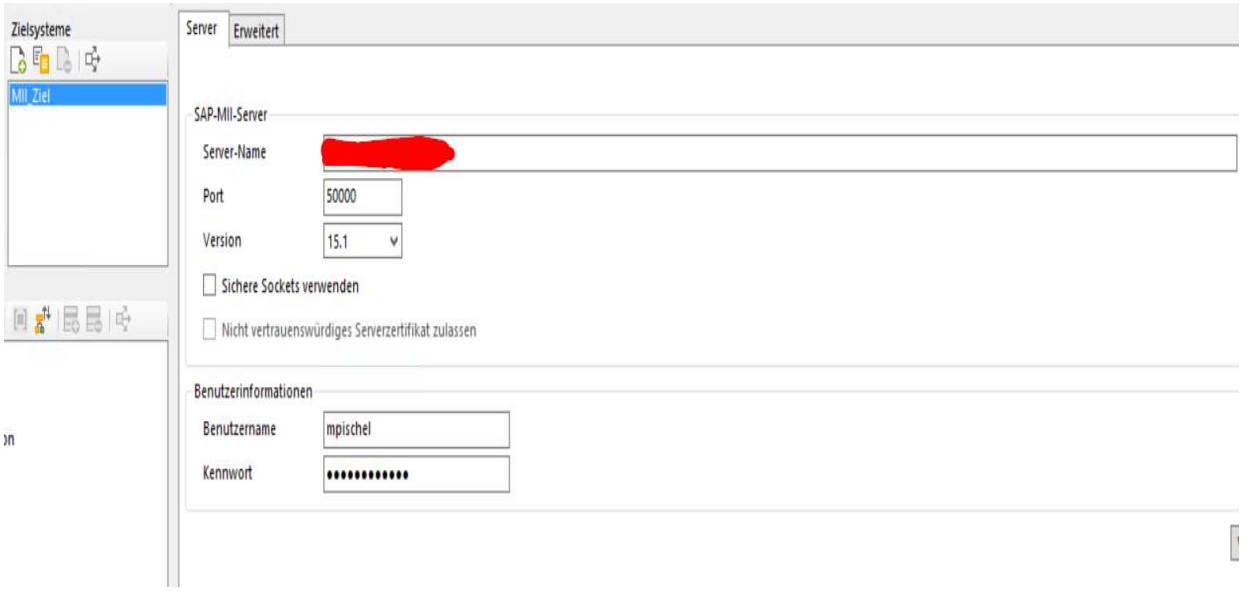

Fig 11. Destination System

A different way of sending the information to the SAP Manufacturing Execution software application is by sending it encapsulated as a Web Service. The PCo software can send the data only by Single Web Service WSDL per defined destination. Even though more than one instance of the destination may be defined, it is possible to instantiate the destination with an Output value to the Web Service discussed.
By using the method of WSDL a different software method can be defined which will actually buffer a hypothetical station in the Plant Connectivity tool.
This buffering will work as following:

- Creating a transaction on the MII side which is used as destination and calls the same WebService
- Buffering started and taking place with the data retrieved from the PLC to the hypothetical station
- Configure the required time for waiting
- Waiting the period of time required for the data to be completed
- Sending forward the data on the required destination.

A different way of sending the information to the SAP Manufacturing Execution software application is by sending it encapsulated as a Web Service. The PCo software can send the data only by Single Web Service WSDL per defined destination. Even though more than one instance of the destination may be defined, it is possible to instantiate the destination with an Output value to the Web Service discussed.
By using the method of WSDL a different software method can be defined which will actually buffer a hypothetical station in the Plant Connectivity tool.
This buffering will work as following:

- Creating a transaction on the MII side which is used as destination and calls the same WebService

- Buffering started and taking place with the data retrieved from the PLC to the hypothetical station
- Configure the required time for waiting
- Waiting the period of time required for the data to be completed
- Sending forward the data on the required destination.

2. CONCLUSIONS

The SAP Plant Connectivity tool is a very useful and powerful tool used in the manufacturing industry nowadays. Speaking of Internet of Things and Industry 4.0 the main need is to move forward for the industry revolution where most of the processes will be automated. Currently, there is a lot of big data passed through this software starting with the processes of beginning and completing a manual/automated operation, overchecking, traceability of big parts, nonconformance requests done by quality agents, start of line/end of line for assembly.

Data is being send as 32 bits character string usually and each bit can relate to a type of processes needed to be triggered or an error that can appear on the operator's screen.

This research was achieved with the support of Technical University of Cluj-Napoca.

3. ACKNOWLEDGMENTS

The authors wish to thank to the Technical University of Cluj-Napoca for the support they offered.

4. REFERENCES

1. Brown, Eric (13 September 2016). "Who Needs the Internet of Things?". Linux.com. Retrieved 23 October 2016.
2. "Internet of Things Global Standards Initiative". ITU. Retrieved 26 June 2015.
3. Nordrum, Amy (18 August 2016). "Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated". IEEE.
4. Hsu, Chin-Lung; Lin, Judy Chuan-Chuan. "An empirical examination of consumer adoption of Internet of Things services: Network externalities and concern for information privacy perspectives". Computers in Human Behavior. 62: 516–527. doi:10.1016/j.chb.2016.04.023.
5. "Internet of Things: Science Fiction or Business Fact?" (PDF). Harvard Business Review. November 2014. Retrieved 23 October 2016.
6. Hendricks, Drew. "The Trouble with the Internet of Things". London Datastore. Greater London Authority. Retrieved 10 August 2015.
7. Ashton, K. (22 June 2009). "That 'Internet of Things' Thing". Retrieved 9 May 2017.
8. "Internet of Things (IoT)". gatewaytechnolabs.com.
9. "The "Only" Coke Machine on the Internet". Carnegie Mellon University. Retrieved 10 November 2014.
10. "Internet of Things Done Wrong Stifles Innovation". InformationWeek. 7 July 2014. Retrieved 10 November 2014.
11. Mattern, Friedemann; Floerkemeier, Christian (2010). "From the Internet of Computers to the Internet of Things" (PDF). Informatik-Spektrum. 33 (2): 107–121. doi:10.1007/s00287-010-0417-7. Retrieved 3 February 2014.
12. Weiser, Mark (1991). "The Computer for the 21st Century" (PDF). Scientific American. 265 (3): 94–104. Bibcode:1991SciAm.265c..94W. doi:10.1038/scientificamerican0991-94. Retrieved 5 November 2014.
13. Raji, RS (June 1994). "Smart networks for control". IEEE Spectrum.
14. Pontin, Jason (29 September 2005). "ETC: Bill Joy's Six Webs". MIT Technology Review. Retrieved 17 November 2013.
15. Analyst Anish Gaddam interviewed by Sue Bushell in Computerworld, on 24 July 2000 ("M-commerce key to ubiquitous internet")
16. Magrassi, P. (2 May 2002). "Why a Universal RFID Infrastructure Would Be

a Good Thing". Gartner research report G00106518.
17. "Peter Day's World of Business". BBC World Service. BBC. Retrieved 4 October 2016.
18. Magrassi, P.; Berg, T (12 August 2002). "A World of Smart Objects". Gartner research report R-17-2243.
19. Commission of the European Communities (18 June 2009). "Internet of Things — An action plan for Europe" (PDF). COM(2009) 278 final.
20. Wood, Alex (31 March 2015). "The internet of things is revolutionizing our lives, but standards are a must". The Guardian.
21. "From M2M to The Internet of Things: Viewpoints From Europe". Techvibes. 7 July 2011. Archived from the original on 24 October 2013.
22. Jürgen Jasperneite:Was hinter Begriffen wie Industrie 4.0 steckt in Computer & Automation, 19 December 2012 accessed on 23 December 2012
23. Kagermann, H., W. Wahlster and J. Helbig, eds., 2013: Recommendations for implementing the strategic initiative Industrie 4.0: Final report of the Industrie 4.0 Working Group
24. Heiner Lasi, Hans-Georg Kemper, Peter Fettke, Thomas Feld, Michael Hoffmann: Industry 4.0. In: Business & Information Systems Engineering 4 (6), pp. 239-242
25. BMBF-Internetredaktion (21 January 2016). "Zukunftsprojekt Industrie 4.0 - BMBF". Bmbf.de. Retrieved 2016-11-30.
26. "Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution". Vdi-nachrichten.com (in German). 1 April 2011. Retrieved 2016-11-30.
27. "Securing the future of German manufacturing industry : Recommendations for implementing the strategic initiative INDUSTRIE 4.0 : Final report of the Industrie 4.0 Working Group" (PDF). Acatech.de. Retrieved 2016-11-30.

CONECTIVITATEA DE PLAN SAP

SAP plan conectivitate (PCo) este un cadru, set de servicii şi instrumente de management care permite schimbul de date între un sistem SAP şi sursele de date standard specifice domeniului diferiţilor producători, cum ar fi sisteme, dispozitive de control, istoric sistemelor plane, sisteme de CPS, TCP prize şi SAP fabricarea produselor (SAP EWM ME, MII de SAP, SAP, SAP ODA şi aşa mai departe). Cu această aplicaţie companii pot primi Tag-uri şi evenimente de la sistemele conectate sursă în producţie automat sau la cerere şi le transmite conectate sistemele SAP. În plus, folosind PCo, produce poate executa, primi, şi procesul de apeluri de metoda OPC UA şi astfel, de asemenea, harta DE sarcini complexe de coordonare în mediul (PLC) Controlere logice programabile. Cercetarea este efectuată cu sprijinul universităţii tehnice din Cluj-Napoca.

Dan BIZUBAC, Phd. Student Eng., Technical University of Cluj-Napoca, Department of Manufacturing Engineering, Muncii Blvd 103-105, Tel: 0040.2640.0641, Cluj-Napoca, ROMANIA,.
Marcel Sabin POPA, Prof. Dr. Eng., Technical University of Cluj-Napoca, Department of Manufacturing Engineering, Muncii Blvd 103-105, Cluj-Napoca, ROMANIA.