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, manufacturers 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.

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.

1.3. Subscription Items

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

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.Overview.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

- **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.

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

15. Analyst Anish Gaddam interviewed by Sue Bushell in Computerworld, on 24 July 2000 ("M-commerce key to ubiquitous internet")
a Good Thing”. Gartner research report G00106518.
20. Wood, Alex (31 March 2015). "The internet of things is revolutionizing our lives, but standards are a must". The Guardian.

CONECTIVITATEA DE PLAN SAP

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