SAP Plant Connectivity Remote Monitoring
How to install, configure and use SAP Plant Connectivity Remote Monitoring Tools

Applies to:
SAP Plant Connectivity 2.3SP5 and later
SAP Plant Connectivity 15.0 and later

Summary
Install, Configure, and connect SAP Plant Connectivity Remote Client to provide remote monitoring of Plant Connectivity systems. Configure Active Monitor to call an MII Transaction on a PCo Agent Instance Failure. Review of the Plant Connectivity ManagementHost Web Service interface.

Author(s): Steve Stubbs

Company: SAP Americas, Inc.

Created on: 05 January 2015

Author Bio
Stephen (Steve) Stubbs of SAP Labs has a Bachelor's Degree in Electrical Engineering Technology from Georgia Southern University. He is a member of the SAP LoB Manufacturing Organization in the Responsive Manufacturing Solution Management group headed by Mike Lackey.
Table of Contents

Overview ........................................................................................................................................... 3
Remote Client ...................................................................................................................................... 3
  Installation ......................................................................................................................................... 3
    As part of a Plant Connectivity installation .................................................................................... 3
    Stand Alone Installation ................................................................................................................ 4
Remote Client Configuration ................................................................................................................ 5
  Adding a PCo System ....................................................................................................................... 5
  Monitoring a Remote System ......................................................................................................... 7
  Remote System Management Selections .......................................................................................... 8
  Remote System Agent Instance Management Options Menu ......................................................... 9
Active Monitoring ............................................................................................................................ 10
  Setting up the MII Transaction .................................................................................................... 11
  Configuring Active Monitoring .................................................................................................. 12
ManagementHost Integration with MII ............................................................................................. 13
  Accessing PCoManagement Web Service from MII ....................................................................... 13
    PCoManagement Web Service Operations ................................................................................... 14
    MII Monitoring Use Cases ......................................................................................................... 16
Related Content ............................................................................................................................... 17
Copyright .......................................................................................................................................... 18
Overview
SAP Plant Connectivity (PCo) provides several mechanisms for remote monitoring and access, and also exposes remote access and control Web Services that can be leveraged by SAP MII or 3rd Party Web Service clients to remotely administer PCo. This article will review the following features for:

- SAP PCo Remote Client
- SAP PCo Active Monitoring
- SAP PCo ManagementHost Web Services integration

All examples and screen shots are from the current shipping version of PCo, 15.0. This information is also valid for PCo 2.3SP5

Remote Client
Remote Client is delivered with the standard PCo installer. Installation can be included with a standard PCo installation, or as a stand-alone install on a remote instance from PCo.

Installation
As part of a Plant Connectivity installation
When selecting the installation options, also make sure that Remote Client is selected under Additional Components:
Stand Alone Installation

For Stand-alone installation, a slightly different installer selection is required. In order to make sure that all of the support components get installed, at least one PCo Agent or Destination should be installed on the remote instance. In this case, select the Simulator Destination from Destinations along with Remote Client from Additional Components:
Remote Client Configuration

Remote Client is a Windows desktop application that can be launched from the Start Menu. PCo server systems have to be added to a Remote Client in order to manage the PCo instance. The Active Monitor service can also monitor both the PCo instance and specific Agent Instances.

Adding a PCo System

Before adding a PCo System, make sure that the PCo System’s ManagementHost Service is started in Automatic or Automatic – Delayed Start on that instance, as seen from the Windows Control Panel Services applet:
After starting Remote Client, right click on *SAP Plant Connectivity Systems*, and select **Add to Monitoring**.

There are two options to add a system: Host Name and SLD (System Landscape Directory from Solution Manager). In this example we will add a system using Use Host Name:

Enter the fully qualified domain name of the PCo Instance Server, set Windows and Use SSL for the Security Settings, and then click **Get System Information**. If a connection is successfully made, you will get a response similar to that shown below:

Click on **Add System** to add this system to the Remote Monitor.
Monitoring a Remote System.

Expanding a Remote System node will show a list of all the configured Agent Instances, and a visual status of each Agent Instance:

- Agent Stopped
- Agent Starting or Stopping
- Agent Running
- Agent Faulted

The status of the ManagementHost connection is shown for the Remote System – Green is connected, Grey is Unknown or not connected.
Remote System Management Selections

Right-click on a system to see the Management Options Menu:

- Delete – Deletes the System from Remote Client
- Manage – Attempts to open a Windows RDP session with the System
- Send Configuration – transfers a PCo Configuration file (exported from PCo in xml format) to the system
- Send Configuration (Overwrite) – same as Send Configuration but Overwrites the existing configuration at the System
- Enable Active Monitoring – See Active Monitoring Below
- Disable Active Monitoring
Remote System Agent Instance Management Options Menu

Right-click on an Agent Instance to see the Management Menu:

- **Start** – Starts a PCo Agent Instance if not running
- **Stop** – Stops a running PCo Agent Instance
- **Restart** – Stops and Restarts a running PCo Agent Instance
- **Retrieve Configuration** – retrieves the PCo Agent Instance Configuration (xml format) for the selected Agent Instance.
- **Retrieve log** – retrieves the PCo Agent Instance Log
- **Enable Active Monitoring** – Enables Active Monitoring, refer to the next section.
- **Disable Active Monitoring** – Disables a configured Active Monitoring session.
Active Monitoring

Setting up Active Monitoring for the Remote Client is a two-step process: Setting up MII Transaction(s) and then configuring the Active Monitoring. As an example, we will use a simple MII transaction that just sends an Email to a user when the Active Monitoring Event is triggered. The event passes XML data to the MII Transaction and the transaction emails the XML to a user.

*Note: Active Monitoring for a PCo Agent Instance currently only monitors the Agent Instance for a **Faulted** State. In order to perform more detailed monitoring of Agent Instance States see the section on ManagementHost Integration with MII.*
Setting up the MII Transaction

- Configure the MII Mail connector
  - From the MII Portal, Select Data Servers, Connections, and create Mail connection to an SMTP Mail server:

  ![Mail connection settings](image)

- Create an MII Transaction that executes the Send_Mail action using the defined Mail connection above.
  - Create a Transaction Input of type Xml to receive the notification from the Active Monitor.
  - Send_Mail action block sends this Xml in the body of the email.
Configuring Active Monitoring

We can monitor both Agent Instance and the PCo ManagementHost service with Active Monitoring.

- **Instance Monitoring**
  - Right Click on an Agent Instance and select **Enable Active Monitoring**. A dialog box will appear that requests connection information:
  - Enter the fully qualified name of the MII Server, the Port on which MII listens for connections, User name and Password, and click **Test Connection** to confirm the connection. Then click **Refresh** to retrieve the Transaction Details. Navigate the Project Tree to select a project folder; then select the Transaction and Input Parameter name of the transaction. A completed dialog is shown here:
  - Click OK to accept and enable the monitor.

- **ManagementHost Monitoring**
  - Right-click on a PCo System name and select Enable Active Monitoring. Perform the same steps to configure the Active Monitor as described above for Instance Monitoring.

After monitoring is enabled, when the Agent Instance goes into Faulted state an email is received with the contents of the XML data sent to MII:

```xml
<?xml version="1.0" encoding="UTF-8"?><pcoAgentInstance name="KepDAAgentInst" node="usphlvm1378" state="Faulted" utctimestamp="1/5/2015 11:01:20 AM"/>
```

If the ManagementHost service is stopped on the host, the XML content returned is:

```xml
<?xml version="1.0" encoding="UTF-8"?><pcoNode name="USPHLVM1378" utctimestamp="1/5/2015 11:20:53 AM"/>
```

For each Agent Instance being monitored, a message is also sent:

```xml
<?xml version="1.0" encoding="UTF-8"?><pcoAgentInstance name="KepDAAgentInst" node="usphlvm1378" state="Unknown" utctimestamp="1/5/2015 11:20:53 AM"/>
```
ManagementHost Integration with MII

The PCo Management Host service (PCoManagment) exposes a number of Web Service operations that can be executed from MII via the Web Service Action block. These operations allow a more detailed monitoring and control of PCo from within MII or any other Web Service client.

Accessing PCoManagement Web Service from MII

By default, the PCoManagment web service listens on port 50050 on the PCo server. The user credentials that are used to connect to the web service must be Windows user login credentials. It is recommended that an MII Credential Alias be created to contain the login information.

From an MII Web_Service Action Block, the Object configuration looks like this:
PCoManagement Web Service Operations

- StartAgentInstance – Starts an Agent Instance by Agent Instance Name
- StopAgentInstance – Stops an Agent Instance by Agent Instance Name
- ReStartAgentInstance – Restarts an Agent Instance by Agent Instance Name
- GetAgentInstanceState – Returns the current State of the Agent Instance by Agent Instance Name. The valid states returned for a configured Agent instance are:
  - Unknown
  - Starting
  - Started
  - Stopping
  - Stopped
  - Faulted
  - StartedWithException
  - ReestablishingConnection

There are other states that are defined for an agent instance that is not configured or being configured, but they should not be seen in a productive system.

- GetAgentInstanceMetaData – Returns a list of all the defined Agent Instances in an xml structure. Each agent instance is defined in the following xml node:

  ```xml
  <a:AgentInstanceMetadata>
    <a:Description/>
    <a:EndpointUrls xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
      <b:string>mii.tcp://usphlvm1535:33000/</b:string>
      <b:string>lisa.tcp://usphlvm1535:33001/</b:string>
    </a:EndpointUrls>
    <a:Name>HDAAgentInst</a:Name>
  </a:AgentInstanceMetadata>
  
  Where
  - Description is an optional agent instance description value
  - EndpointUrls is a list of the ports on which the agent instance listens for a defined Query Port configuration
  - Name is the Name of the instance.

- GetSingleAgentInstanceMetaData – Returns a single agent instance metadata in the same xml structure as defined for GetAgentInstanceMetaData above
- GetPCoMetaData – returns the following metadata xml structure:

  ```xml
  <?xml version="1.0" encoding="UTF-8"?>
  <GetPCoMetadataResponse xmlns="urn:sap.com:pco.management">
    <GetPCoMetadataResult xmlns:a="urn:sap.com:pco.contracts" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <a:Caption>PCo Instance on usphlv1535</a:Caption>
      <a:Hostname>usphlv1535</a:Hostname>
      <a:HttpEndPointUrl>http://usphlv1535:50050/PCoManagement</a:HttpEndPointUrl>
      <a:Name>PCO</a:Name>
      <a:NetTcpEndPointUrl>net.tcp://usphlv1535/PCoManagement</a:NetTcpEndPointUrl>
      <a:ProductName>Plant Connectivity</a:ProductName>
      <a:SecureHttpEndPointUrl i:nil="true"/>
      <a:SoftwareComponentVersion>2.305.2141.1051</a:SoftwareComponentVersion>
    </GetPCoMetadataResult>
  </GetPCoMetadataResponse>
  ```
Where

- Caption is a description string
- Hostname is the name of the Windows Server
- HttpEndPointUrl is the http endpoint for the PCoManagment web service
- Name is the PCo System Id
- NetTcpEndPointUrl is the .Net endpoint for the PCoManagment web service
- ProductName is the PCo product name
- SecureHttpEndPointUrl is a placeholder
- SoftwareComponentVersion is the current full software revision for PCo

- **GetAgentInstanceConfiguration** – Returns a specified Agent Instance’s base64-encoded zGip content containing the Agent Instance Configuration xml.
- **GetAgentInstanceConfigurationWithKey** -- Returns a specified Agent Instance’s base64-encoded zGip content containing the Agent Instance Configuration xml. The xml is secured with a string key value specified in the request.
- **SendAgentInstanceConfiguration** – sends an Agent Instance Configuration xml document to PCo using the following inputs:
  - configXml – contains the XML document
  - key – the security key used to secure the xml document
  - overwrite – set true to overwrite the current configuration, false to create a new agent Instance.
- **SendAgentInstanceConfiguration2** – sends an Agent Instance Configuration xml document to PCo using the following inputs:
  - configXml – contains the XML document
  - key – the security key used to secure the xml document
  - overwrite – set true to overwrite the current configuration, false to create a new agent Instance.
  - importChangedConfig – set true to accept a configuration xml that has been modified from the original export and has a different checksum than the original export xml.
- **GetAgentInstancesRuntimeInfo** -- returns an XML response with a node for each Agent Instance:

```
<a:PCoRuntimeInfo>
  <a:Name>KepDAAgentInst</a:Name>
  <a:Status>Started</a:Status>
</a:PCoRuntimeInfo>
```

Where

- Name – Agent Instance Name
- Status – Agent Instance State

- **GetAgentInstancePagingLog** -- returns an XML representation of a specified Agent Instance Log using the following inputs:
  - instance – Agent Instance Name
  - start – Start Date of the log data
  - end – end Date of the log data
  - traceEventType – numeric log severity level (this level and all levels <= than this level)
    - Critical==1
    - Error==2
    - Warning==4
    - Information==8
    - Verbose==16
  - startindex – offset of the rows in the selection (defaults to 0)
  - pagesize – number of rows to return
The XML document contains an Items node list with an AgentLogItem node for each row of the log, as shown in this example:

```xml
<a:AgentLogItem>
  <a:Datetime>2015-01-06T16:30:56.4259618-05:00</a:Datetime>
  <a:EventId>1809</a:EventId>
  <a:Machine>USPHLVM1378</a:Machine>
  <a:Message>Sending method 'doIsConnected' to STA thread for execution.</a:Message>
  <a:ProcessId>9000</a:ProcessId>
  <a:ProcessName>PCoSvcHost</a:ProcessName>
  <a:Source>StaThread</a:Source>
  <a:StackTrace/>
  <a:ThreadId>30</a:ThreadId>
  <a:TraceEventType>Verbose</a:TraceEventType>
</a:AgentLogItem>
```

**MII Monitoring Use Cases**

Some use cases for monitoring of PCo Agent Instances are:

- Detecting when the Source Agent connection is unavailable
  - Call `AgentInstancesRuntimeInfo` from a Scheduled transaction and check for `State=ReestablishingConnection`

- Detecting when the Agent Instance is not Started
  - Call `AgentInstancesRuntimeInfo` from a Scheduled transaction and check for `State<>Started`

- Restarting a failed or non-Started Agent Instance
  - Call `StartAgentInstance` or `ReStartAgentInstance`

- Monitoring the functioning of PCoManagement
  - Call `GetPCoMetaData` and test for a valid response
Related Content

PCo Component Overview
New MII PCoQuery Fixed Query Mode
SAP Manufacturing Implementation Architecture
Copyright

© 2014 SAP SE or an SAP SE 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.

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 and its affiliated companies (“SAP SE Group”) for informational purposes only, without representation or warranty of any kind, and SAP SE Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE Group 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.

SAP SE and other SAP SE products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE in Germany and other countries.