Integrate IBM Predictive Maintenance and Quality (PMQ) with ILS deviceWISE to onboard high-value asset data

Identify problem hot spots and their solutions

Seba Kauser (seba_kauser@in.ibm.com)   Application/System Integration Specialist
IBM

Vignesh Velusamy Ravindran (vrvignesh@in.ibm.com)   Solution Architect
IBM

Randy Giffen (randy_giffen@ca.ibm.com)   Architect
IBM

06 November 2014

High-value asset-based industries need to be aware of the performance of their components in a real-time mode. Handle any future breakdowns or failures, and control the impact to your organization's financial performance. Feed data from level 2 systems and historians in real time to build this awareness and control. Learn to integrate IBM® Predictive Maintenance and Quality with ILS deviceWISE to receive data from remote sources.

Overview

Asset-intensive industries, such as oil and gas, mining, and energy and utilities, use complex equipment, such as compressors, haul trucks, and turbines, in their day-to-day operation. Any unplanned downtime or major unforeseen failure of this equipment has a direct impact on production downtime, which affects the financial performance of the organization.

Potential component and equipment failure, plus machine health of in-service equipment needs to be monitored by identifying early signs of possible downtime. The goal is to maximize the uptime of the component/equipment.
The IBM Predictive Maintenance and Quality (PMQ) solution helps you monitor, analyze, and report on information that is gathered from high-value assets and recommend maintenance activities for them. With this integrated solution, you can:

- Predict the failure of a monitored asset in order to fix it and avoid costly downtime.
- Search stored maintenance logs to determine the best repair procedures and cycles.
- Identify the root causes of asset failure to take corrective actions.

The integration bus layer within PMQ helps to transform external events (received from monitored high-value assets) into the format that is required by the PMQ analytics solution’s data model. One way to receive external low-level events, such as the discharge pressure of a compressor or the inlet temperature of a compressor, is to use the ILS deviceWISE Machine to Machine (M2M) Application Platform. (For more information about ILS deviceWISE, see Resources.) This platform helps:

- Connect assets to applications.
- Collect and process data from a variant of assets, in various locations.
- Integrate that data into existing enterprise IT systems or to a custom dashboard to drive better business decisions.
- Remotely access and manage assets.

This article explains how to onboard data from an OLE for Process Control (OPC) source, which stands for Object Linking and Embedding (OLE) for Process Control, and the step-by-step configuration to onboard data into PMQ to perform analytics on the asset data.

**Use case**

A typical oil and gas industry involves various rotating equipment, such as turbines, pumps, compressors, generators, and motors. Each gas turbine has an upstream rotating compressor, which generates pipeline data, such as compressor unit suction and discharge pressures, gas temperatures, unit flow, ambient temperature, and more. The data that is received from low-level source systems acts as a data store to perform analytics, like the health monitoring of turbines. The data is captured in the form of events from field-level systems in a real-time mode that uses a protocol and format that is called OLE for Process Control (OPC). OPC event sources are integrated by using an external adapter, ILS deviceWISE, which pushes the data into the Integration Bus layer within PMQ by using the WebSphere® MQ (IBM MQ) transport. After device events are received in a queue, the event-processing component transforms them into the PMQ event format that is required by the solution.

Tags are often used in the process industry and are normally assigned to a piece of information. A tag consists of a name by describing a single point of information, so a process system can consist of hundreds and even thousands of tags.

For demonstration of integration capabilities, this article uses MatrikonOPC Simulator, which provides equipment data connectivity by using these tags, which are an item database for a device. Simulated tags are OPC-timestamped with range and OPC quality. Based on the quality set, a tag can either be "good" or "bad."
In this scenario, each compressor carries about nine tags that indicate different measurements, as shown in Table 1.

### Table 1. Compressor tags, with their units of measurement

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>English unit</th>
<th>Metric unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTIM</td>
<td>Compressor inlet temperature</td>
<td>Fahrenheit (°F)</td>
<td>Celsius (°C)</td>
</tr>
<tr>
<td>CTD</td>
<td>Compressor discharge temperature</td>
<td>Fahrenheit (°F)</td>
<td>Celsius (°C)</td>
</tr>
<tr>
<td>CPR</td>
<td>Compressor pressure ratio</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>AFPCS</td>
<td>Compressor inlet pressure transducers 96CS</td>
<td>in H2O</td>
<td>mm H2O</td>
</tr>
<tr>
<td>CPD</td>
<td>Compressor discharge press max select</td>
<td>psi</td>
<td>bar</td>
</tr>
<tr>
<td>AFQ</td>
<td>Compressor inlet air mass flow</td>
<td>lb/s</td>
<td>kg/s</td>
</tr>
<tr>
<td>CompEff_Mean</td>
<td>Mean compressor efficiency</td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>AFPAP</td>
<td>Barometric pressure transducers 96AP</td>
<td>in</td>
<td>mm</td>
</tr>
<tr>
<td>CMHUM</td>
<td>Specific humidity</td>
<td>Ratio</td>
<td>Ratio</td>
</tr>
</tbody>
</table>

### Solution architecture

Figure 1 shows the integration of ILS deviceWISE with IBM Predictive Maintenance and Quality solution, a packaged, preconfigured cross-industry business analytics solution.

**Figure 1. Architecture of the solution when you integrate ILS deviceWISE with PMQ**
Technical summary of the solution workflow

1. Integration with historians: Equipment tags and online analyzers are configured in DCS, SCADA, and historian systems. ILS deviceWISE is configured to read the values of all the online and offline tags in a real-time mode. Thus, ILS deviceWISE integrates with level 2 systems and historians to fetch sensor, alarm, monitoring, and diagnostic data that is injected from equipments. This integration results in continuous raw event data capture and a near real-time analysis of the captured data.

2. Real-time data integration: Triggers are created in ILS deviceWISE workbench to put the tag data in the integration bus layer within PMQ by using a message queue. This data is received on the queue in the form of XML.

3. Onboard operational data store: Event data that is received from historian systems is converted to the format expected by PMQ, which is eventually populated in the analytical data store. This onboarded data is aggregated data and includes key performance indicator (KPI) and profile information.

4. Predictive model: IBM SPSS® executes pre-built analytical models, resulting in scores. KPIs are analyzed by the system on a continuous basis. In response to the scores and the current KPI values, SPSS generates recommendations by using the pre-configured business rules.

5. Enterprise Asset Management (EAM) systems: The received recommendation can be used to initiate or modify a work order in EAM (Maximo) systems for maintenance of the compressor. This event also provides an automatic email alert in all such instances.

High-level configuration to integrate IBM PMQ with ILS deviceWISE

- Configure MatrikonOPC to simulate tag values for a compressor
- Configure ILS deviceWISE
- Configure IBM PMQ
- Test the solution

Configure MatrikonOPC to simulate tag values for a compressor

Create an alias group in the MatrikonOPC Simulation Server, with the required alias for each parameter, as shown in Figure 2. (For more information about MatrikonOPC, see Resources.)

1. Open the Matrikon OPC Server for Simulation.
2. Right-click on Alias Configuration and select Insert Alias Group from the pop-up menu.
3. Provide a suitable name for the alias group.
4. In the Contents of alias group frame for the newly created group, right-click and select Insert New Alias from the pop-up menu.
5. Provide a suitable name for the alias/parameter.
Configure ILS deviceWISE

1. Create a device of type DA CLIENT in ILS deviceWISE and specify the required OPC server URL. (The OPC server URL field differs per your setup. It is not a generic URL to be used by all.)

   **Figure 3. Device of type DA Client, pointing to the correct OPC server**

After the device is connected, it displays the items in the Variable tab as in Figure 4.
Figure 4. DA Client with a list of available alias groups and their aliases that are configured in Matrikon OPC

2. Create a WebSphere MQ transport and provide the required queue manager, queue, channel, and host name.

Figure 5. WebSphere MQ transport that is created in deviceWISE

3. Create a transport map for each tag by selecting the previously created IBM MQ transport. Repeat this step for each tag.
Figure 6. Transport map

Figure 7 provides a screen capture of the XML message that is generated.

**Figure 7. XML message that is generated after the map is successfully created**

4. Create a project within the node, as shown in Figure 8.
   a. In the NEW NODE tree in deviceWISE, expand NEW NODE.
   b. Right-click on the projects item, and select **New** from the menu.
c. Provide a new project name in the wizard.  
A new project then appears in the Projects tab, as shown in Figure 8.

Figure 8. A new project that is created under Node section in deviceWISE

<table>
<thead>
<tr>
<th>Name</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAM</td>
<td>Started</td>
</tr>
<tr>
<td>PAMC</td>
<td>Started</td>
</tr>
<tr>
<td>PAMC_Comp2</td>
<td>Stopped</td>
</tr>
<tr>
<td>PAMC_Lock_Comp3</td>
<td>Stopped</td>
</tr>
<tr>
<td>PAMC_Lock_Comp4</td>
<td>Started</td>
</tr>
<tr>
<td>PAMC_Lock_Comp5</td>
<td>Started</td>
</tr>
</tbody>
</table>

5. Create a trigger for each tag within the project by using the Canvas editor.

Figure 9. Canvas editor, depicting a new trigger for a tag

After the trigger definition is validated and saved, it appears within the project, along with all the other triggers, as shown in Figure 10.
Integrate IBM Predictive Maintenance and Quality (PMQ) with ILS
deviceWISE to onboard high-value asset data

Figure 10. Project with a list of all available triggers in deviceWISE workbench

```
NEW NODE / Projects
Projects
Name     | Location | State  | Status | Last Triggered | User | Successes | Failures
AFPC5C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC5C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC4C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC3C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC2C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC1C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC0C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC9C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC8C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC7C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC6C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC5C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC4C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC3C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC2C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC1C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC0C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC9C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC8C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC7C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC6C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC5C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC4C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC3C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC2C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
AFDC1C_Trigger | Data     | Stopped | Unloaded | nyel            | NA   | NA        | 0
```  

Configure IBM PMQ

Master data is the type of resource that you want to manage, such as people, parts, assets, pieces of equipment, and processes. Master data is normally loaded by using one of the supplied connectors or the Flat File API. The connectors and the Flat File API use IBM Integration Bus flows to transform the data into the required form and to update the data in the IBM Predictive Maintenance and Quality database.

Explore master data and other concepts in the PMQ solution guide (see Resources).

The following master data files must be loaded in PMQ data store to populate the master tables:

- location_upsert.csv
- measurement_type_upsert.csv
- profile_variable_upsert.csv
- resource_upsert.csv
- source_system_upsert.csv

The resource master data sheet contains a list of all compressors and their attributes, as shown in Figure 11.

Figure 11. Sample master data sheet for a resource in IBM PMQ
Test the solution

Start the trigger that is created in ILS deviceWISE.

1. Right-click on the trigger present in a project under 'Node' in deviceWISE.
2. Select Start to start the trigger.

Figure 12. Starting trigger in ILS deviceWISE workbench

Starting the trigger sends a message into the queue that was defined in the transport section. The message is of the format defined in the transport map definition. Data that is pushed from ILS is received in XML in a WebSphere message queue.

You can browse for the XML message received from ILS.

Figure 13. List of messages that are received in WebSphere MQ

You can double-click any one of the XML messages to see a detailed view of the data present in the XML, as shown in Figure 14.

Figure 14. Detailed view of XML message
The event data that is received from historian systems is converted to the format expected by PMQ, which is eventually populated in the analytical data store. This onboarded data is aggregated data and includes KPI and profile information.

You can see a screen capture of the event observation table in PMQ data store, as shown in Figure 15.

**Figure 15. Data onboarded in the PMQ data store**

![Data onboarded in PMQ data store](image)

**Conclusion**

The IBM Predictive Maintenance and Quality solution helps you monitor, analyze, and report on information that is gathered from devices and other assets and recommend maintenance activities. PMQ uses ILS deviceWISE to integrate seamlessly with level 2 systems and historians to perform predictive and business analytics on the operational data. This analysis provides the hot spot identification of a problem and the corresponding resolution to avoid a forced outage.
Resources

Learn

• "Real-time data analytics using IBM Predictive Maintenance and Quality" (developerWorks, May 2014): Understand how to use IBM PMQ to onboard production data in real time and perform analytics on the data to predict production in near future.
• IBM Predictive Maintenance and Quality Information Center: Learn more about the solution in the IBM Predictive Maintenance and Quality Information Center.
• "Predictive Maintenance and Quality 1.0 Solution Guide" (IBM, 2013): Gain an understanding of how the IBM Predictive Maintenance and Quality solution works. Know what tasks are involved when you plan to implement IBM Predictive Maintenance and Quality. (The solution asset that is used in this article is based on PMQ 1.0.)
• "IBM Predictive Maintenance and Quality 2.0 Solution Guide" (IBM Redbooks, May 2014): Learn how Predictive Maintenance and Quality enables companies to identify when manufacturing assets need maintenance, not just according to the manufacturer’s scheduled repair guide but also based on how the asset is used every day. This information helps to keep critical production lines running while also saving money because repairs are always, and only, performed when truly necessary.
• "Predict the future to keep your production line running" (IBM): View a demo on how IBM Predictive Maintenance and Quality helps spot problems before they happen so you can plan for, rather than react to asset failure.
• ILS deviceWISE: Learn more about ILS deviceWISE and how to seamlessly connect your assets with your enterprise systems and databases.
• MatrikonOPC: Learn more about MatrikonOPC and MatrikonOPC Simulation Server.

Get products and technologies

• IBM Predictive Maintenance and Quality solution: Explore the IBM Predictive Maintenance and Quality solution, which helps you maximize asset productivity and operational performance.
• Evaluate IBM products in the way that suits you best: Download a product trial, try a product online, use a product in a cloud environment.

Discuss

• PMQ Practitioners Community: Get involved in the PMQ Practitioners Community to share knowledge, ideas, solutions, and experiences around IBM Predictive Maintenance and Quality.
• Get involved in the developerWorks community. Connect with other developerWorks users while you explore the developer-driven blogs, forums, groups, and wikis.
About the authors

Seba Kauser

Seba Kauser works with Global Business Services at IBM India and is part of the Predictive and Business Analytics Industry Solution and Services group. Working with the chemical and petroleum community, she uses the IBM Predictive Maintenance and Quality solution to meet the requirements of chemical and petroleum customers. Her interests include application integration and middleware that use WebSphere products in various industry domains.

Vignesh Velusamy Ravindran

Vignesh Velusamy Ravindran is a solution architect with the Industry Analytics Solution and Services team in India. In this current role, he works with clients to implement predictive analytics solutions that are based on IBM analytics platform. Before this role, he worked with the IBM business partner’s team, where he led various partner engagements to design and develop SOA-based applications that use the WebSphere portfolio. He has more than 15 years of IT experience.

Randy Giffen

Randy Giffen is an Architect with the Business Analytics Solution team in Ottawa, Canada. Previously, he was a member of the WebSphere Connectivity and Process Management Tooling team, and before that he worked on the Eclipse project. He has 17 years of experience with software development at IBM.

© Copyright IBM Corporation 2014
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)