DPM Solution Benchmark

Version 1.2

PTC Inc., 121 Seaport Boulevard, Boston, MA 02210 USA

Document Revision History

Revision Date	Version	Description of Change
April 22, 2022	1.0	Document creation
May 11, 2022	1.1	Updated document links to new DPM Help Center
June 15, 2022	1.2	Clarified about bandwidth limitations

Introduction

"*Digital Performance Management (DPM) is a closed-loop, problem solving solution that helps manufacturers identify, prioritize, and solve their biggest loss challenges, resulting in reduced cost, increased revenue, and improved service levels.*"

– DPM Help Center

What is DPM?

Digital Performance Management (DPM) is an application which improves factory efficiency across a variety of different areas, namely "the four P's" of Digital Transformation: products, processes, places, and people. Each performance issue in a factory can be mapped to at least one of these improvement categories in a new strategy for Continuous Improvement (CI) founded by PTC.

Continuous Improvement Focus Areas[11]	**Problem Identified with Digital Performanced Management**
Products; processes; places	Unplanned downtime
Processes; people	Changeovers
Processes; people	Speed
Products; processes	Scrap

Figure 1 – Each performance issue in a factory can be mapped to at least one of 4 fundamental improvement categories: products, processes, places, and people. PTC's new, industry-leading strategy for continuous improvement (CI) in factories is a "best practice" approach, taking the collective knowledge of many customers to form a focused, prescriptive path for success.

11 Closing the Loop Across Products, Processes, People, and Places, Manufacturing Leadership Journal

At PTC, CI in factories is driven by a "best practice" approach, with years of experience in manufacturing solutions combining with the collective knowledge of the many diverse use cases PTC has encountered, to generate a focused, **prescriptive** path for improvement in any individual factory. PTC is also defining new industry standards for OEE analysis by using time as a currency within DPM. This standardization technique improves intuitive impact assessment and allows for direct comparison of metrics (see the Help Center for details on how each metric is calculated).

DPM creates a closed loop for CI, from the monitoring phase performed both automatically and through manual operator input, to the prioritization and analyzation phases performed by plant managers.

Figure 2 – DPM is a closed loop for continuous improvement, a strategy built around industry standard best practices and years of experience.

DPM helps plant managers by tracking metrics of factory performance that often go overlooked by other systems. With Analytics, DPM can also do much of the analysis automatically, finding the root causes much more rapidly.

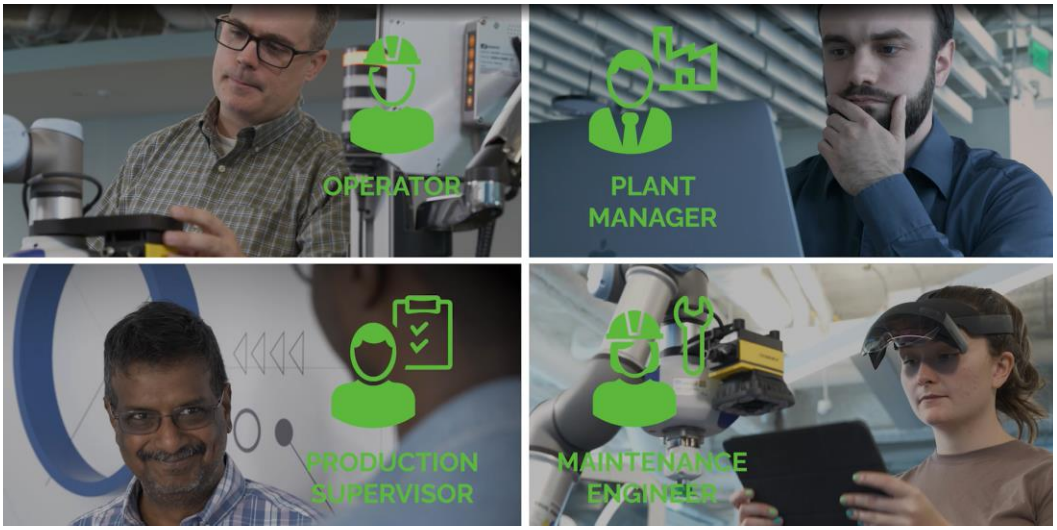
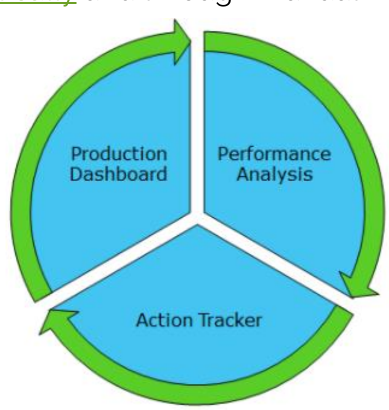


Figure 3 – All levels of the company are involved in solving the same problems effectively and efficiently with DPM. Instead of 100 people working on 100 different problems, some of which might not significantly improve OEE anyway, these same 100 people can tackle the top few problems one at a time, knocking out barriers to continuous improvement together.

Production supervisors who manage the entire production line then know which less-than-effective components on the line need help. They can quickly design and redesign solutions for **specific** production issues. Task management within DPM helps both the production manager and the maintenance engineer to complete the improvement process. Using other PTC tools like Creo and Vuforia make the path to improvement even faster and easier, requiring less expert knowledge from the front-line workers and empowering every level of participation in the digital transformation process to make a direct, measurable impact on physical production.

How Does DPM Work?

DPM as an IoT application sits on top of the ThingWorx Foundation server, a platform for IoT development that is extensible and customizable. Manufacturers therefore find they rarely have to rip and replace existing systems and assets to reap the benefits of DPM, which gathers, aggregates, and stores production data (both automatically and through manual input on the Production Dashboard), so that it can be analyzed using time as a currency. DPM also manages the process of implementing improvements (using the Action Tracker) based on the collected data, and provides an easy way to confirm that the improvements make a real difference in the overall OEE (through the Performance Analysis Dashboard). Because the analysis occurs before and after the steps to improve are taken, manufacturers can rest assured that any resources invested on the improvements aren't done so in vain; DPM is a predictive and prescriptive analysis process.



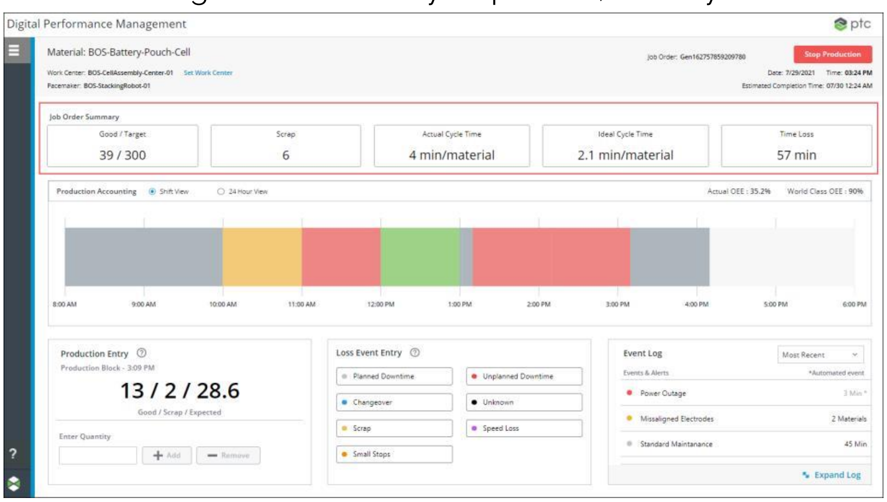
DPM makes use of an external SQL Server to run queries against collected data and perform aggregation and analysis tasks in the background, on a separate server location than the thing model and ingestion database. This ensures that use cases involving real-time alerts and events, high-capacity ingestion, or others are still possible on the ThingWorx Foundation server.

The IoT EDC is focusing in on DPM alone for a series of documents which detail the expected performance of DPM in an average factory environment. This will serve as a baseline for ensuring your DPM application ease-of-use and improving chances of adoption amongst your factory staff.

The DPM User Experience

DPM is a tool designed to be beneficial at all levels of a company, from the operators monitoring automated data on production events from the factory machines themselves, to the production supervisors who need to establish, task out, and track machine maintenance and improvement measures. DPM also engages the continuous improvement and plant leadership, by providing a standardized way to monitor performance that ultimately rolls up to the executive level. The end users of DPM are therefore diverse both in how they access DPM, and how they make use of its various features.

One of the perks to building DPM on top of the ThingWorx Foundation is that many of the webpages (called "mashups") within ThingWorx are already responsive, and any which aren't responsive OOTB can be modified and custom designed for different size viewing screens to ensure that if necessary, end users can access DPM from a variety of locations and devices. Most of the time, end users will be accessing mashups from hard-wired dashboards mounted on the actual devices, or from wireless laptops which have standard size screens with standard resolutions. For use cases involving phones or tablets, however, it may be necessary to see how DPM will perform across a variety of bandwidth and latency conditions. Often, cellular or satellite connection is a must to facilitate field team cooperation, and 5G networks often result in worsened performance.



So, to demonstrate the influence of bandwidth and latency on the responsiveness of DPM, the Production Dashboard was loaded in the Google Chrome browser repeatedly under varying conditions. This dashboard is the webpage most operators and field users would access to log event information and production details (so it is widely used by end users).

Methods

Latency was introduced by hosting the servers involved in the test in different regions (all Azure cloud hosted servers, one in US East, one US West, and one in Japan East). Bandwidth was introduced using a tool on the PC with either no bandwidth or 4 megabits/second.

Browser caching was turned on and off as well, to simulate the difference between new and return users; new users would not have the webpage cached, so their load times are expected to be longer. Tomcat compression was also configured in half of the runs to demonstrate the importance of compression for optimal performance.

Each of these 24 scenarios was then tested 10 times from each location. Provided are the load times reported by Developer Tools in Google Chrome.

Bandwidth and Latency Test Runs

Users Access DPM From Servers Located in the/a…

Same Region as the Hosted Application

Azure US East - DPM Production Dashboard Load Times (in Seconds, 10 tests performed per row)								
Client Bandwidth Limit	Client Browser Cache	Server Compression	Minimum	Median	Maximum	Standard Deviation	Average	User
None	ON		2.3	2.4	2.8	0.2	2.5	Return
None	ON	ON	2.3	2.5	2.9	0.2	2.6	Return
None			2.5	2.6	2.8	0.1	2.6	New
None		ON	2.8	3.2	3.6	0.3	3.2	New
4 mbit/sec	ON	ON	2.5	2.7	2.9	0.1	2.7	Return
4 mbit/sec	ON		4.0	4.2	4.4	0.1	4.2	Return
4 mbit/sec		ON	6.4	6.5	6.8	0.1	6.5	New
4 mbit/sec			22.0	22.1	22.2	0.0	22.1	New

Figure 4 – Return users (row 1&2) on good internet (wireless networks within factories) have the best performance. New users (row 3&4) have only slightly slower load times in these common access use cases. Compression matters very little. Slower networks (mobile networks) see slower but still good performance, better for return users, which is good since DPM field users are more likely to be accessing DPM often.

Moderately Distant Region

Azure US West - DPM Production Dashboard Load Times (in Seconds, 10 tests performed per row)								
Client Bandwidth Limit	Client Browser Cache	Server Compression	Minimum	Median	Maximum	Standard Deviation	Average	User
None	ON		4.6	4.7	5.6	0.3	4.8	Return
None	ON	ON	4.7	5.1	6.3	0.4	5.2	Return
None			5.1	5.3	5.4	0.1	5.2	New
None		ON	5.6	5.9	6.3	0.2	5.9	New
4 mbit/sec	ON	ON	4.8	5.0	5.3	0.2	5.0	Return
4 mbit/sec	ON		5.7	5.8	6.6	0.2	6.0	Return
4 mbit/sec		ON	8.9	9.0	9.3	0.1	9.0	New
4 mbit/sec			23.9	24.0	24.1	0.1	24.0	New

Figure 5 – Latency (40-50 ms) was introduced in this scenario by changing regions to a nearby region, like one on the other side of the country. Note in this chart and the previous one the need for compression when mobile networks are expected to be used. Load times nearly double, but the app is still performant enough to be used actively.

Region on the Other Side of the World

Azure Japan East - DPM Production Dashboard Load Times (in Seconds, 10 tests performed per row)								
Client Bandwidth Limit	Client Browser Cache	Server Compression	Minimum	Median	Maximum	Standard Deviation	Average	User
None	ON		7.4	7.8	8.9	0.4	7.9	Return
None	ON	ON	5.7	6.0	6.3	0.2	6.0	Return
None			7.9	8.3	12.1	1.2	8.7	New
None		ON	6.7	6.9	8.7	0.6	7.2	New
4 mbit/sec	ON	ON	5.6	5.8	6.2	0.2	5.8	Return
4 mbit/sec	ON		7.9	8.2	8.3	0.1	8.1	Return
4 mbit/sec		ON	10.0	10.1	10.5	0.1	10.1	New
4 mbit/sec			20.8	26.0	26.8	1.6	25.6	New

Figure 6 – The same trends continue with more latency from a region much farther away (150-160 ms). Notice how return mobile network users even on the other side of the world have better performance than new mobile users in the same region. Return performance matters most of all when considering ease-of-use of the DPM application.

Key Takeaways

Latency and bandwidth impact DPM performance in exactly the way one would expect of a web application. While any DPM server can be accessed from any region, regions with more latency will experience delays proportional to the amount of latency. In the chart here, find the three regions represented three times by three different colors (different from the charts above):

- The **three different shades** of each color represent the **different regions**
- Green represents the optimal configuration settings (**Tomcat compression enabled, caching turned on**) for returning users with bandwidth limitations (i.e. mobile networks like 5G)
- Blue shows first-time page visitors with no bandwidth limitations
- Purple shows first-time visitors that do have bandwidth limitations
- The uncompressed first-time load for mobile users (those with bandwidth limitations imposed) within the same region is also given to demonstrate the importance of enabling Tomcat Compression (load times only get worse without compression the farther the region)

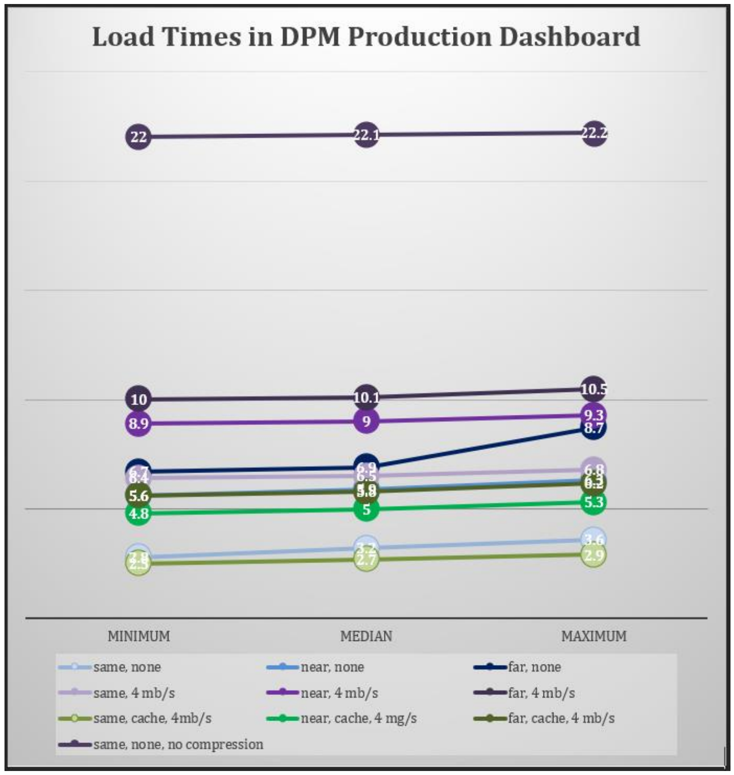


*Figure 7 – Even with bandwidth limitations, every region sees better performance for return users versus new users, which may be important to note. However, because DPM field users most commonly access DPM often, the return user time is a better indicator of adoption, and those numbers look great in our simulations. Notice the top line which shows the very worst of mobile performance, what happens over networks with limited bandwidth when **Tomcat Compression** is not enabled. Load times vary only slightly for regular networks when Tomcat Compression is enabled, and they vastly improve performance across regions and on mobile networks, so it's **highly recommended** (instructions are below).*

Notice how the green series has lower load times across the board than the blue one, meaning that **return users even with bandwidth limitations have better performance across every region than new users**. Also notice how the gap is larger between lighter colors and darker colors, where the darker the color, the farther the region from the DPM servers. **This indicates that network latency has a more significant influence on performance versus bandwidth**, with only longer running transactions like file uploads seeing a significant performance hit when on a network with bandwidth.

Enabling Tomcat Compression

Note that Tomcat compression can be enabled by adding the below highlighted options to the Tomcat *server.xml* file:

```
<Connector port="8080" protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443"

    compression="on"
    compressionMinSize="2048"
    noCompressionUserAgents="Thingworx"
    compressableMimeType =
"text/html,text/xml,text/javascript,application/javascript,text/css,text/plain,
text/json,application/json"
/>
```

Appendix: DPM References

Topic Link		Detailed Summary of Content
"the four P's"	1.0	Areas to focus on in order to improve factory efficiency: **products, processes, people, and places**. "As both digital performance management and these applications are built upon an IoT platform, they don't require manufacturers to rip and replace existing systems and assets [to solve a variety of problems]."
DPM Requires SQLServer	1.0	Much of the DPM tool relies on stored procedures in SQL Server, which means that only MSSQL or PostgreSQL ThingWorx can use DPM, and a database username and password must be manually configured for DPM as part of the installation process.
DPM Deployment in HA	1.0	As with any ThingWorx Extension, DPM should be deployed into a scaled-down single node, and then the cluster scaled back up to improve performance across the entire HA deployment.
Calculations (OEE, etc.)	1.0	The way in which OEE, Effective Time, Planned Production, and other metrics of production efficiency are calculated can be found in this reference. Notice how every metric is calculated around time as a currency, a standardization technique which improves intuitive impact assessment and allows for direct comparison of all metrics.
Timezones in DPM		Timestamps in the database are all UTC, and timestamps in the views found throughout DPM are all in local time (to the viewer).
Licensing in DPM	1.0	DPM requires a special license be installed before it can be imported.
Glossary of DPM Terms	1.0	All DPM terms defined formally.