**USE CASE USER MANUAL**

|  |  |
| --- | --- |
| **Use Case Title: DPM Extended** | |
| **Use Case version: 9.5** | **Release date: Jan 23 2024** |
| **Author: Andrew Kharma** | **Email: akharma@ptc.com** |
| **Job title: Senior Validation Engineer** | **Company: PTC** |

**Chapter 1 - USE CASE OVERVIEW**

DPM Plus contains a group of customizations that solve common problems found across multiple customers. The following list details each customization:

* Multi-Machine Management (MMM)
  + A customization that allows for data entry at every work unit as well as handling multiple pacemakers working in parallel. Use this solution to:
    - Collect data at every work unit both manually and using automation
    - Mark multiple work units as pacemakers working in parallel
* Status Timeline Chart
  + A customization that displays a timeline of the status of the machine on the Production Dashboard
  + Used either with Automation or with the Manual Availability Events customization
* Manual Availability Events
  + A customization that allows the operator to enter specific start and end times for loss events
  + Used in conjunction with the Status Timeline Chart customization
* Editable Automation Events
  + A customization that allows the user to modify automation events to change the reason, comment, and work unit causing event
* Enable/Disable Metrics
  + A customization that allows the user to enable or disable certain metrics so that they do not appear in the Scorecard List
* New Custom Metrics
  + A customization that adds many new metrics to the Scorecard view including APQ and Loss Percentages
* New Custom Reports
  + A customization that adds new reports to DPM including the Shift Summary, Summary Dashboard, and Time Loss Pareto reports
* New Action Tracker Task Types
  + A customization that adds new task types to action tracker including the ability to create ServiceMax Work Orders and Navigate Problem Reports

**Chapter 2 - USE CASE - BLOCK ARCHITECTURE**

* PTCDTS.OperationKPIImpl
  + Implementation of the DPM Building Block 'PTC.OperationKPIImpl'
* PTCDTS.ProductionDashboard'
  + Implementation of the DPM Building Block 'PTC. ProductionDashboard'
* PTCDTS.ScorecardImpl'
  + Implementation of the DPM Building Block 'PTC. Scorecard'
* PTCDTS.KPIAnalysisImpl'
  + Implementation of the DPM Building Block 'PTC. KPIAnalysisImpl'
* PTCDTS.WorkDefinitionExecutionResponse
  + New Database Building Block that creates and manages a new DB table
* PTCDTS.TimeLossImpl'
  + Implementation of the DPM Building Block 'PTC. TimeLossImpl'
* PTCDTS.PerformanceAnalysis'
  + Implementation of the DPM Building Block 'PTC. PerformanceAnalysis'
* PTCDTS.Administration
  + New UI Building Block to add a new administration mashup
* PTCDTS.BottleNeckImpl'
  + Implementation of the DPM Building Block 'PTC. BottleNeckImpl'
* PTCDTS.ActionTrackerImpl
  + Implementation of the DPM Building Block 'PTC. ActionTrackerImpl
* PTCDTS.Reports
  + New Standard Building Block
* PTCDTS.SMAXIntegration
  + New Standard Building Block

**Chapter 3 - SOFTWARE VERSIONS**

Thingworx version: 9.5

PTC common blocks version: 9.5

**Chapter 4 – DEPENDENCIES**

|  |  |
| --- | --- |
| Building Blocks | DPM 9.5 |
| Extensions | N/A |
| Solutions/Other | N/A |
| Widgets | N/A |

**Chapter 5 – INSTALLATION**

The following steps walk through how to install the Solution Framework:

* Download the extension archive
* Log into ThingWorx Composer as an Administrator that has access to import extensions
* In the left-side navigation menu, click on the Import/Export button (Diverging Arrows Icon)
* Click the Import Sub-Menu
* In the popup that displays, change Import Option to Extension
* Click the Browse button and navigate to the location the downloaded <FILE\_NAME>.zip file
* Select the zip archive and click the open button
* Click the Import button. A confirmation should appear that the import was successful. Close the popup.
* Navigate to PTC.Base.Manager
* Click on the Services tab.
* Click the 'Execute' button next to the 'InitializeSolution' service
* Run with the following JSON Structure:

*{*

*"databaseUser": "<TWX\_DB\_USER\_NAME>",*

*"databasePassword": "<TWX\_DB\_USER\_PASSWORD>",*

*"databaseJDBCString": "<TWX\_DB\_JDBC\_STRING>",*

*"databaseThing": "PTC.DBConnection.MSSQLDatabase or PTC.DBConnection.PostgresDatabase"*

*}*

**Chapter 6 – CONFIGURATION**

Once installed, an admin will need to configure the system's master data so that each work unit has a work master for every material created in that line. This is most easily done using the data import tool provided [here](https://github.com/PTCInc/solution-data-import).

The excel sheet needs to explicitly define work masters at each work unit.

* In the WorkMaster tab (may be hidden):
  + A work master needs to be defined at the work center level for each material.
  + A work master needs to be defined at all work units for each material
* In the WorkMasterLink tab (may be hidden):
  + Every work unit work master needs to be linked to the appropriate work center work master.
* In the WorkMasterModelSpecification tab (may be hidden):
  + Each work master needs to be linked to its associated work unit or work center
* In the WorkMasterMaterialSpecification tab (may be hidden):
  + Each work master needs to be linked to its associated material.
  + The quantity column in this tab can be used to define ratios between the number produced by the work center and each work unit
    - For example, if the work center's quantity is set to 1 and a work unit's quantity is set to 2, when a job order is stated on that work unit, the target quantity for the work unit will be 2 times the target specified by the job.
* In the WorkMasterConfigurationSpec tab (may be hidden):
  + Each work master must be linked to one of the pacemakers in the associated work center.
* The WorkMasterData tab needs to be deleted
* The WorkCenter tab needs each row to have the Base Template field set to PTCDTS.OperationKPIImpl.DefaultWorkCenter\_TT or another template that is derived from it
* The WorkUnit tab needs each row to have the Base Template field set to one of the following (or another template that derives from them)
  + PTCDTS.OperationKPIImpl.DefaultWorkUnit\_TT: for standard work units without automation
  + PTCDTS.OperationKPIImpl.AutomationEventsWorkUnit\_TT: for work units that will use automation data

Other Configurations:

* If gathering data at non-pacemakers is not required, you can set the DisplayNonPacemakers configuration on PTCDTS.OperationKPIImpl.Manager to false (defaults to true).
  + This will hide non-pacemakers in the dropdown on Production Dashboard
* If you wish to disable the Status Time Line or Manual Availability Events functionality, you can do so by updating the enabled field on the appropriate row of the CustomizationConfiguration config table in PTCDTS.ProductionDashboard.Manager
* For Manual Availability Events, if you wish to automatically create an event when manually starting/stopping production on a work unit, you can update the StartStopJobOrderReasonMapping configuration on PTCDTS.ProductionDashboard.Manager to denote which reasons you wish to use.
* Enabling or Disabling Metrics can be done by toggling the isEnabled field of the appropriate row in the MetricConfiguration config table in PTCDTS.ScorecardImpl.Manager
* Enabling or Disabling task type can be done by toggling the enabled field on the TaskTypeConfiguration table in PTCDTS.ActionTrackerImpl.Manager

**Chapter 7 – USAGE**

The Production Dashboard can be used normally with a few key differences:

* Work Unit selection is no longer automatic. The user can select which work unit to work on
* When stopping production on a work unit, the option exists to force a stop on the job order. Set this to true if you want to change the state of the job order regardless of the state of other work units.
  + If this is not set to true, the Job Order Status will only change if all other work units are in the same state.
* If Status Timeline is enabled, the user can now toggle between a Production Block or Status view to see the Status Timeline.
* If Manual Availability Events is enabled, the user will now be prompted to input a start and end time whenever they enter a Down time event. Scrap and Speed Loss events will be entered normally.

The Waterfall Chart can be used normally with the added fact that the user can select a work unit from the equipment drop down.

The Scorecard view will not show new metrics:

* Planned Downtime (%): A percentage of planned downtime vs total time
* Unplanned Downtime (%): A percentage of unplanned downtime vs total time
* Changeover (%): A percentage of changeover vs total time
* Speed Loss (%): A percentage of Speed Loss vs total time
* Small Stops (%): A percentage of Small Stops vs total time
* Availability (%): percent availability of the machine over time
* Performance (%): percent of actual performance vs ideal performance
* Quality (%): Ratio of Good Production to Total Production

Under the new Reports menu, you can see some new reports:

* Time Loss Pareto: A “flattened” view of the waterfall. This will show case your highest impact loss reasons broken down by machine
* Shift Summary: This will display loss and production data grouped by machine, job order, and shift
* Summary Dashboard: This will display job order information like current status, OEE, and more for the selected group of work units