# Oracle® Adaptive Intelligent Apps for Manufacturing

User's Guide

Release 20A

**Part No. E88122-10**

April 2020

ORACLE

Oracle Adaptive Intelligent Apps for Manufacturing User's Guide, Release 20A

Part No. E88122-10



Primary Author: Laurel Dale, Sujata Patnaik

Contributing Author: Mariyappan Arumugham, Swapna Kommineni, Manasi Puranik

# Contents

## 6 Analyzing Data



## 7 Presenting Data



## A Navigation



## B Model Features



## C Algorithms



## D REST Web Services

# Send Us Your Comments

**Oracle Adaptive Intelligent Apps for Manufacturing User's Guide, Release 20A**

**Part No. E88122-10**

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Oracle E-Business Suite Release Online Documentation CD available on My Oracle Support and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: appsdoc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

# Preface

## Intended Audience

Welcome to Release 20A of the *Oracle Adaptive Intelligent Apps for Manufacturing User's Guide*.

See Related Information Sources on page x for more Oracle E-Business Suite product information.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

## Structure

## Related Information Sources

Oracle Big Data Cloud Service [https://docs.oracle.com/en/cloud/paas/big-data-cloud/index.html]

Oracle Cloud Infrastructure Object Storage Documentation [https://docs.us-phoenix-1.oraclecloud.com/Content/Object/Concepts/objectstorageoverview.htm]

Oracle Data Mining Concepts [https://docs.oracle.com/en/database/oracle/oracle-database/18/dmcon/index.html]

Oracle Data Pump [https://docs.oracle.com/en/database/oracle/oracle-database/12.2/sutil/oracle-data-pump.html#GUID-501A9908-BCC5-434C-8853-9A6096766B5A]

Oracle Database Cloud Service [https://docs.oracle.com/en/cloud/paas/database-dbaas-cloud/index.html]

Oracle GoldenGate Cloud Service [http://docs.oracle.com/cloud/latest/goldengate-cloud/index.html]

## Do Not Use Database Tools to Modify Oracle E-Business Suite Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle E-Business Suite data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle E-Business Suite data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle E-Business Suite tables are interrelated, any change you make using an Oracle E-Business Suite form can update many tables at once. But when you modify Oracle E-Business Suite data using anything other than Oracle E-Business Suite, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle E-Business Suite.

When you use Oracle E-Business Suite to modify your data, Oracle E-Business Suite automatically checks that your changes are valid. Oracle E-Business Suite also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

# 1

# Overview

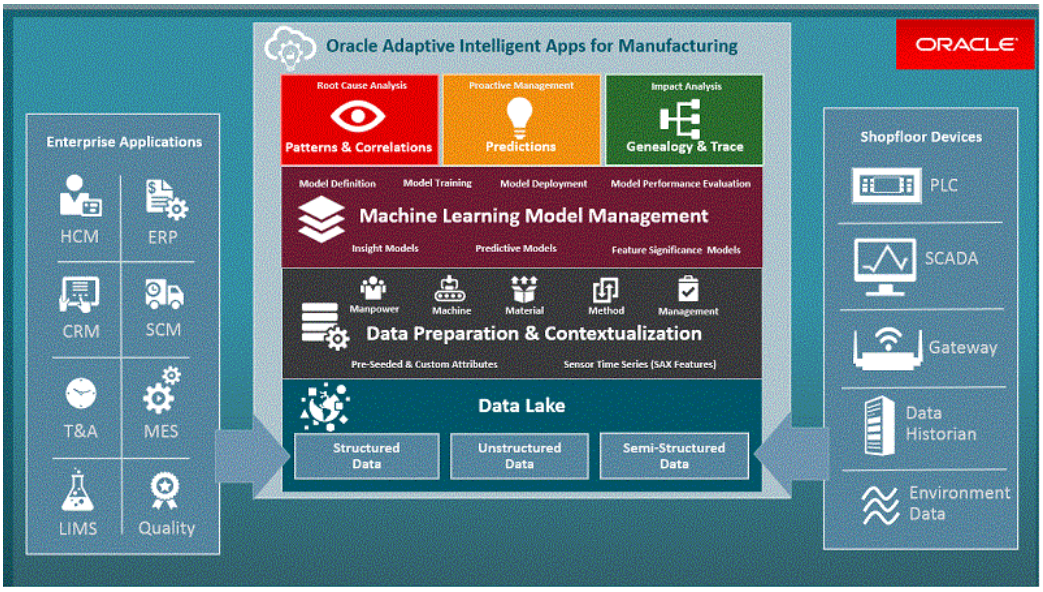
This chapter covers the following topics:

- Overview of Oracle Adaptive Intelligent Apps for Manufacturing

## Overview of Oracle Adaptive Intelligent Apps for Manufacturing

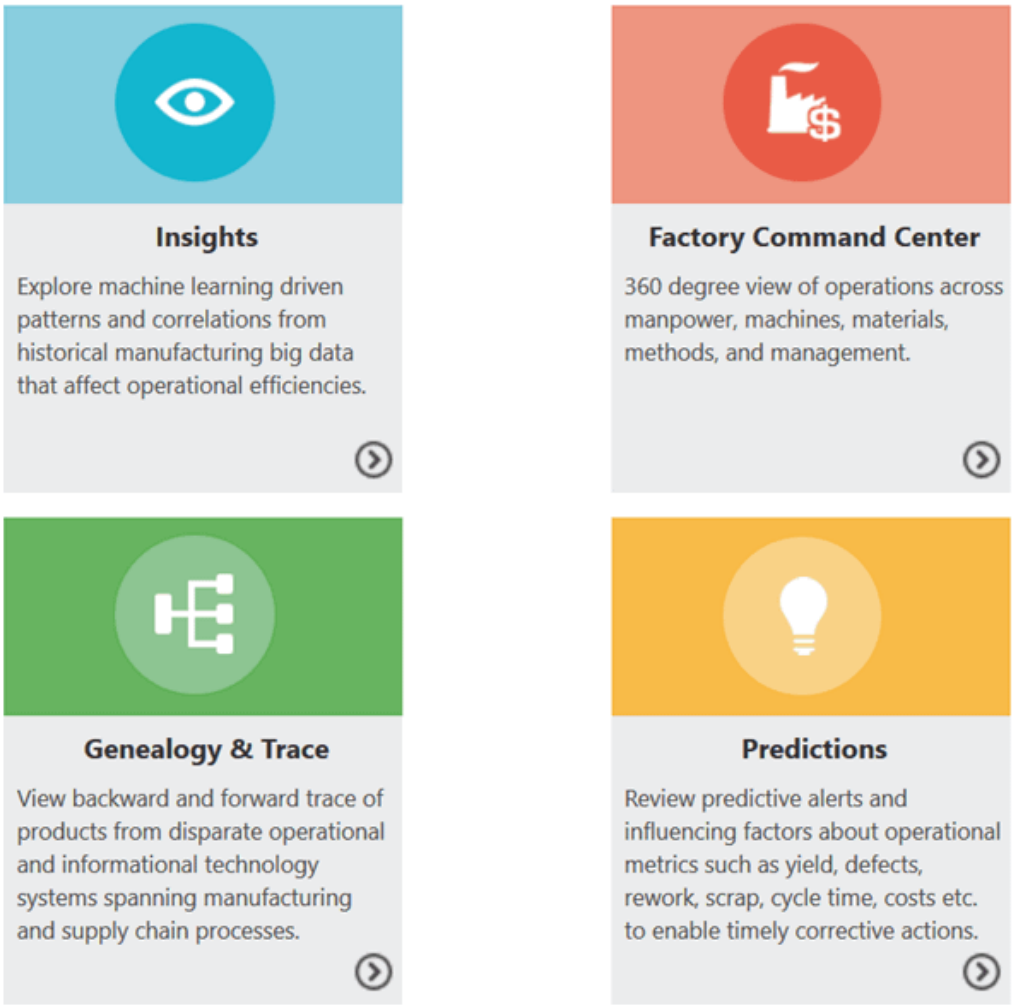
Oracle Adaptive Intelligent Apps for Manufacturing collects, stores, and analyzes massive amounts of Operational Technology (OT) data coming from shopfloor systems such as equipment, machines, sensors, and test stations and then contextualizes it with Information Technology (IT) data coming from business applications such as Supply Chain Manufacturing (SCM), Enterprise Resource Planning (ERP), Human Capital Management (HCM), and Customer Relationship Management (CRM). Oracle Adaptive Intelligent Apps for Manufacturing then analyzes the data by applying machine learning, data mining, and artificial intelligence techniques to discover key patterns and correlations that affect manufacturing efficiencies and provides actionable predictive analytics to maximize yield, and minimize defects, scrap, cycle times, costs, etc. It also provides comprehensive capabilities for backward and forward tracing of products and processes within manufacturing and supply chain spanning manpower, machine, material, method and management aspects to facilitate rapid root cause, impact and containment analysis. The following graphic depicts this flow of data:

***Oracle Adaptive Intelligent Apps for Manufacturing Data Flow***




Oracle Adaptive Intelligent Apps for Manufacturing provides four modules with advanced analytical capabilities. These four modules, which you can access from the Home page, shown in the graphic below, are:

- Insights
- Predictions
- Genealogy and Trace
- Factory Command Center




## Insights

Insights from historical data analysis enable business users to discover the hidden patterns between influencing factors and a production metric, known as the target measure. The insights are presented to the business user in an easy to use user interface. Business users can drill into a specific insight to gain comprehensive understanding about the influencing factors by visualizing the correlations and distributions from the historical data set. Insights are supported for process manufacturing , discrete manufacturing and serialized manufacturing use cases.

Use the Insights module to:

- Create a dataset to prepare the context for analyzing historical data.
- Create an Insights model for a target measure using selected features and algorithms.

- Review and publish the insights.
- Deploy the Insights model.
- View the findings in aesthetic and self-explanatory visualizations.

## Predictions

The Predictions module provides predictive alerts on a target measure in the current or later operations in the production cycle. The module alerts you to variations from the target early in the process based on predictive analysis.

Process Manufacturing users can access real time predictions of a product target measure for current and future operations.

Discrete Manufacturing users can access real time predictions of an assembly target measure for current and future operations.

Additionally, discrete serialized manufacturing users can view real time predictions of an assembly serial unit target measure for current and future operations.

Use the Predictions module to:

- Create a dataset to prepare the context for analyzing historical data.
- Create a Predictions model that helps to predict a target measure for a product using selected predictors and an algorithm.
- Analyze the confusion matrix to evaluate model performance. The confusion matrix displays the accuracy of the model (actual vs. predicted target measure results) for various results classifications, such as high, on target, or low.
- Deploy the Predictions model.
- View predictions for the batches, work orders, or serial units currently in production.

## Genealogy and Trace

The Genealogy and Trace module enables end-to-end tracking and analysis of material composition, test results, and processes undertaken at every operation step of the manufacturing process, which provides complete traceability of a partially or completely finished product. This traceability includes items procured (internally or externally), consumed, manufactured, outsourced, shipped, returned, repaired, serviced in the field, and re-shipped during a particular period.

The Genealogy and Trace module includes a Timeline Viewer and a Network Viewer to enable traceability using a simple and intuitive visualization of the entire product genealogy. Use these viewers to drill down through objects such as equipment, sales

orders, purchase orders, user defined entities, and other details to trace the impact of an object in the supply chain network. You can easily switch back and forth between the Network and Timeline Viewers as needed from any object in the view.

Use the Timeline Viewer to view:

- All events associated with a work order, an equipment instance, a lot number, or a serial number occurring over time.
- Only certain steps or types of events associated with a work order, equipment instance, lot number, or serial number.
- The details of any associated event.

Use the Network Viewer to view:

- A purchase order, lot, serial number, work order, or sales order and the associated entities.
- Details about the relationship between two objects, such as the lot quantity received for a purchase order.
- A serialized item or lot controlled item and then trace the item from the supplier to its consumption in a work order and its shipment to a customer.

## Factory Command Center

The Factory Command Center displays different metrics and prediction alerts related to the overall factory/organization and classified under the 5 Ms of the factory: Manpower, Machine, Material, Management, and Method. Examples of metrics and alerts you can expect to see are:

- Manpower - information related to the operators assigned to the running work orders, such as operators not clocked in, not reported, and skills mismatch.
- Machine - the machines that are down, scheduled for maintenance, and idle.
- Material - material with a shortage or expiring.
- Management - work order-related delays, such as work order start or finish delays, operation delays, or work orders on hold.
- Method - changes to the planned method of operation, such as changes in operation or activity duration, unplanned operations, or material, processing, and resource exceptions.

The above metrics and alerts are reported for current work orders with a status of Unreleased or Pending, Released, In Progress, and On Hold.

# 2

# Managing Security and Access Control

This chapter covers the following topics:

- Managing User Roles and Access
- Creating and Managing Users
- Setting Up Your IDCS User Profile
- Defining User Preferences

## Managing User Roles and Access

Oracle Adaptive Intelligent Apps for Manufacturing uses the following application persona and roles in process or discrete manufacturing to access the four modules, namely Insights, Predictions, Genealogy and Trace, and Factory Command Center, as well as the Setup page:

- Business User
- Data Analyst
- Data Scientist
- Application Administrator

These persona have access to the application modules, features and data relevant to their role. Based on their roles they have access to perform the following functions:

### Metadata Extraction and Data Preparation

The data analyst and data scientist can perform the following functions related to extracting metadata and preparing the data:

- Extract feature metadata to analyze the production quality or yield for a given product, recipe or bills of material (BOM), and routing operation context.

- Use the Key Performance Indicators page to create and manage key performance indicators.
- Use the Time Series Features Sets page to create and manage time series features sets.
- Use the data preparation support for out of box seeded features from structured data (imported through various methods of data ingestion) and configurable Time series data (equipment sensor summaries).
- Use Feature Significance models to find the most significant features, and rank features using the Chi-Square or the Random Forest Classifier algorithms.
- Use Feature Significance model feature rankings information to select features when building Insight and Prediction model definitions.

## Model Building for Predictions and Insights

The data analyst and data scientist can perform the following functions related to model building for predictions and insights:

- Build Prediction and Insight models using features for process or discrete manufacturing.
- Define analysis models using a simple and visually rich wizard.
- Build Insight models using the Apriori algorithm (Association Rules) or the Decision Tree Classification algorithms.
- Build Prediction models using the Support Vector Machine or the Decision Tree Classification algorithms.

## Model Evaluation and Publication

The data analyst and data scientist can perform the following functions related to evaluation and publishing models:

- Evaluate how much each feature influences target measure results. Use the most significant features to create your Insights and Predictions models.
- Evaluate the insight patterns and rules generated and deploy the insights model, enabling collaboration with business analyst and users.
- Evaluate prediction models using the confusion matrix and deploy the right model for production use.

## Predictive Analysis

The business user, data analyst, and data scientist can perform the following functions related to predictive analysis:

- View classified predictions of the KPI targets such as production quality elements, production yield or cycle time at the work order and operation level.
- Comprehensively understand the influencing predictors by visualizing the statistics generated on the training data set (Descriptive Stats, Histogram and Box plot).
- Take corrective actions to improve KPI targets using predictions early in the production cycle.
- Leverage support for the historical data analysis of the following use cases:
    - Analyze target measure of quality, yield, or custom KPI for Process Manufacturing.
    - Analyze target measure of quality, yield, or custom KPI for Discrete Manufacturing.
    - Analyze target measure of quality, yield, or custom KPI for Discrete Serialized Manufacturing.

## Historical Data Analysis

The business user, data analyst, and data scientist can perform the following functions related to historical data analysis:

- Discover hidden patterns between the influencing factors and target measure for quality, yield, or custom KPIs.
- Visualize insights in an easy-to-understand format.
- Comprehensively understand influencing factors by visualizing the correlations and distributions with the target measure.
- Leverage support for the historical data analysis of quality, yield, or custom KPIs in the following manufacturing scenarios:
    - Process manufacturing
    - Discrete manufacturing
    - Discrete serialized manufacturing

## Factory Command Center

The business user and data analyst can monitor the following functions using the Factory Command Center:

- Monitor the 5M facets of the factory: Manpower, Machine, Management, Material, and Method from a single page.
- Track key metrics for each category.
- Monitor and resolve alerts, exceptions and errors as they occur in near real time.
- Access details (like equipment, work orders, and items) underlying each category, evaluate the status (such as machine down, work order delayed, and operator unassigned) and the supporting numbers and reasons.
- View prediction alerts for the monitored entities. Navigate to the predictive analysis module to further evaluate the prediction and underlying statistics.

## Genealogy & Trace

The business user and data analyst can perform the following functions related to genealogy and trace:

- View time lines of:
    - a work order to review work order operations, operators, process exceptions, components used, quality test results, machine alerts and parametric data related to the machine.
    - a machine to review work order exceptions, maintenance events and schedules, machine alerts and parametric data related to the machine.
    - a serial unit to review serial unit from purchase or production to consumption or sale showing the lineage of the serial unit from its creation to final consumption in the supply chain.
    - a lot to review the lot creation transaction, expired lot and failed quality inspection alerts, the item, and the lot's status.
- Access details of every event and zoom in and out of time periods.
- View complete genealogy of lots and serials and easily traverse complex networks by highlighting key paths and nodes and controlling their visibility and explosion.
- Navigate back and forth between networks and time lines like for example navigate to the genealogy of a serial controlled component from the timeline of the work

order in which the serial unit was executed.

- Use the Network Viewer to visualize the genealogy from the perspective of the lot or serial or by purchase order, production work order, sales order, or user defined entity.

## Administration

The applications administrator can perform the following functions:

- Use the Organization Access page to provide:
  - Access to multiple organizations for a user.
  - Access for multiple users to an organization.
  - Access to the Create Organization page.
- Use the Background Process page to submit a new request.

## REST APIs

To use REST web services, a user must have access to Oracle Identity Cloud Service with either the Identity Domain Administrator or Application Administrator role. The user must first register the AIAMFG Client Application in Oracle Identity Cloud Service, page D-7, then use a third-party client such as Postman to access REST web services, page D-8.

**Warning:** REST web services do not verify a user's organization access.

## User Roles and Access

The following table shows the access to the modules and pages based on user roles:

| Adaptive Intelligent Apps For Manufacturing Menu | Roles and Access | | | |
|---|---|---|---|---|
| | **Business User** | **Data Analyst** | **Data Scientist** | **Application Administrator** |
| Home | Yes | Yes | Yes | Yes |

| | | | | |
|---|---|---|---|---|
| **From the Home page**: | | | | |
| Insights | Yes | Yes | Yes | Yes |
| Factory Command Center | Yes | Yes | No | Yes |
| Genealogy & Trace | Yes | Yes | No | Yes |
| Predictions | Yes | Yes | Yes | Yes |
| Setup | No | No | No | Yes |
| **From the Insights page**: | | | | |
| Quality | Yes | Yes | Yes | Yes |
| Yield | Yes | Yes | Yes | Yes |
| Serial Quality | Yes | Yes | Yes | Yes |
| Serial Yield | Yes | Yes | Yes | Yes |
| Evaluation | No | Yes | Yes | Yes |
| Modeling | No | Yes | Yes | Yes |
| Data Preparation | No | Yes | Yes | Yes |
| Data Ingestion | No | Yes | Yes | Yes |
| Configuration | No | Yes (but not Time Series Feature Sets) | Yes | Yes |
| **From the Genealogy & Trace page**: | | | | |

| | | | | |
|---|---|---|---|---|
| Timeline Viewer | Yes | Yes | No | Yes |
| Network Viewer | Yes | Yes | No | Yes |
| **From the Predictions page**: | | | | |
| Quality | Yes | Yes | Yes | Yes |
| Yield | Yes | Yes | Yes | Yes |
| Serial Quality | Yes | Yes | Yes | Yes |
| Serial Yield | Yes | Yes | Yes | Yes |
| Confusion Matrix | No | Yes | Yes | Yes |
| Modeling | No | Yes | Yes | Yes |
| Data Preparation | No | Yes | Yes | Yes |
| Data Ingestion | No | Yes (but not Sensor Devices Data) | Yes | Yes |
| Configuration | No | Yes (but not Time Series Feature Sets) | Yes | Yes |
| **From the Setup page**: | | | | |
| Organization Access | No | No | No | Yes |
| Key Performance Indicators | No | No | No | Yes |
| Time Series Feature Sets | No | No | No | Yes |

| | | | | |
|---|---|---|---|---|
| **From the User Name menu**: | | | | |
| Preferences | Yes | Yes | Yes | Yes |
| Background Processes | No | No | No | Yes |
| Profile | Yes | Yes | Yes | Yes |

For complete navigation details, see: Oracle Adaptive Intelligent Apps for Manufacturing Navigation, page A-1

## Creating and Managing Users

AIAMFG uses Oracle Identity Cloud Service to manage users and provide access to AIAMFG. When your organization signs up for an Oracle Cloud account and then orders subscriptions to cloud services, including AIAMFG, your organization designates one person to receive an email containing a link used to activate the services ordered. See: Getting Started with Oracle Cloud [https://docs.oracle.com/en/cloud/get-started/subscriptions-cloud/csgsg/index.html]

Your Oracle Cloud account includes Oracle Identity Cloud Service (IDCS) as well as AIAMFG. See: Oracle Identity Cloud Service [https://docs.oracle.com/en/cloud/paas/identity-cloud/index.html]. The person who activates the services ordered is assigned the Identity Domain Administrator role by Oracle and can create other users with various roles. See: Managing Oracle Identity Cloud Service Users [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/manage-oracle-identity-cloud-service-users1.html]. The user with the Identity Domain Administrator role can manage AIAMFG users themselves or they can create one or more users with the IDCS User Administrator role to add the AIAMFG users and with the IDCS Application Administrator role to grant the AIAMFG application and roles to users.

The newly created user receives an email with a link to activate their Oracle Cloud account. Once the new user activates their account and they have been assigned a role for an application instance, then they can log in to the system and define their user preferences. Each user must specify their preferred organization and category set, at a minimum. See Defining User Preferences, page 2-15.

Your Identity Domain Administrator applies default settings to the entire identity domain in the cloud. If the identity domain encompasses multiple time zones and languages, for example, you may prefer to specify personal settings. AIAMFG supports multiple languages. To select one of these languages, see Setting Up Your IDCS User Profile, page 2-12.

### To add a new user:

Either a user with the IDCS Identity Domain Administrator role or the IDCS User Administrator role can add a new user.

1. Navigate to Oracle Identity Cloud Service. Click your name icon, then **Admin Console**.

2. In Oracle Identity Cloud Service, add a user. Refer to Creating User Accounts [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/create-user-accounts.html] for the remaining steps.

3. The user receives an email, asking them to activate their account. The user must click the link provided.

   The user still requires access to an application and an application role, as described in the next steps, before they can use the account.

### To assign a user to an application

Either a user with the IDCS Identity Domain Administrator role or the IDCS Application Administrator role can assign a user to an Oracle application and application role.

4. Navigate to Oracle Identity Cloud Service. Click your name icon, then **Admin Console**.

5. In the Identity Cloud Service console, expand the Navigation Drawer, click **Oracle Cloud Services**, and then click your application, named AIAMFG_<Instance_Type>.

   **Additional Information:** Oracle Identity Cloud Service can have multiple application instance types. Examples of instance types include production (with an application name of AIAMFG_PROD) and test (with an application name of AIAMFG_TEST).

6. Refer to Assigning Users to Oracle Applications [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/assign-users-oracle-applications.html] for the remaining steps.

   Choose from the following roles:

   - Application Administrator
   - Data Analyst
   - Data Scientist

- Business User

### To assign organization access to a user:

Only a user with the AIAMFG Application Administrator role can assign organization access to a user.

1. Navigate to the Organization Access page.

   From the Home page, click **Setup**, then **Organization Access**.

2. In the Organization Access page, use the **Person Name** search field to search for a user.

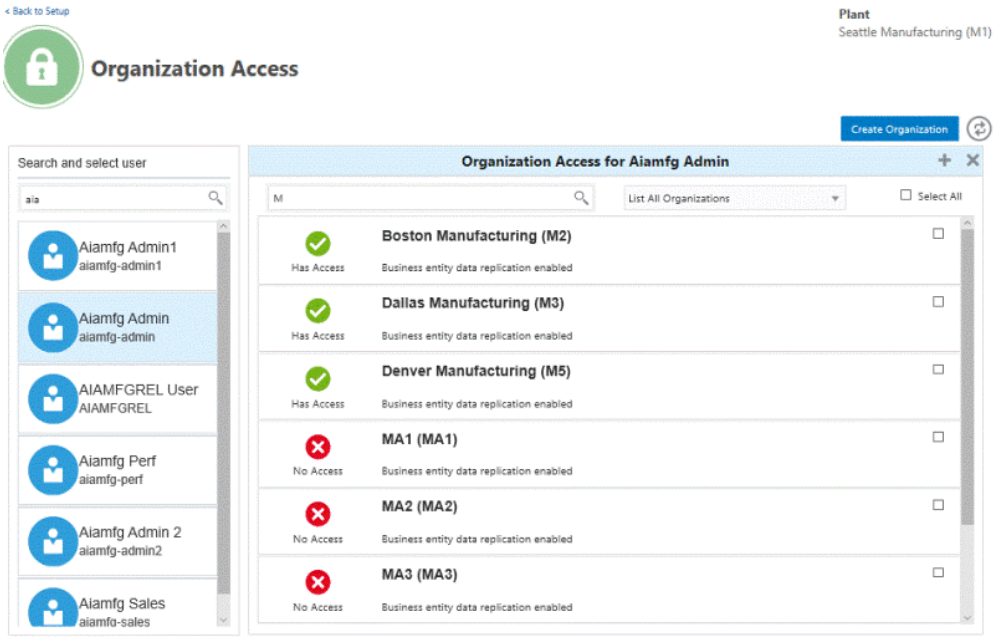
3. Use the **Org Code, Name** search field to search for an organization.

   **Tip:** You can also narrow the list of organizations by selecting:

   - List all organizations
   - List organizations with user access
   - List organizations with no user access

4. Select the check box for the organization.

   **Tip:** Select multiple organizations to update them all for the same user at the same time.




5. Click the plus icon to assign organization access to the user or click the X icon to remove organization access.

   A green check mark appears beside the organization name once user access is assigned. A red X appears beside the organization name to indicate no user access.

   **Tip:** Using the steps above, you can add or remove organization access to multiple organizations at one time for one user. Alternatively, use the Swap button (to the right of the Create Organization button) to update multiple users for one organization at the same time.

### To create a data ingestion organization:

As Oracle Adaptive Intelligent Apps for Manufacturing displays analysis data by organization, the data collected from various data sources using different ingestion methods belongs to a unique organization code. You must define and be assigned to a Case Record Data Ingestion organization to upload case record data files and a Business Entity Data organization to upload the business entity data files.

To create a Case Record or Business Entity data ingestion organization, see Setting Up Data Ingestion, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

## Related Topics

Removing User Accounts [https://docs.oracle.com/en/cloud/paas/identity-

cloud/uaids/remove-user-accounts.html]

Assigning Groups to Oracle Applications [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/assign-groups-oracle-applications.html]

Removing Groups from Oracle Applications [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/remove-groups-oracle-applications.html]

## Setting Up Your IDCS User Profile

Your IDCS domain has user profile settings that you can choose to personalize, such as a preferred language and your login password. AIAMFG supports the following languages, in addition to the default language, English (United States):

- Dutch
- French
- French (Canada)
- German
- Italian
- Japanese
- Korean
- Portuguese (Brazil)
- Russian
- Spanish
- Latin American Spanish (includes all variants of Spanish used outside of Spain)
- Turkish

IDCS by default shows all languages, but AIAMFG supports only a subset of these languages. AIAMFG selects a language for translation of the user interface in the following order, until it finds a supported language. AIAMFG selects:

1. The language in the My Profile Details tab of the My Profile console in IDCS, unless this language is not supported. See: Set Up or Modify Your Profile [https://docs.oracle.com/en/cloud/paas/identity-cloud/usids/set-or-modify-your-profile.html] or To set up your IDCS user profile, page 2-14.
2. The supported base language of a selected variant language, unless this base language is not supported. For example, if you select Dutch (Netherlands), the

variant language, AIAMFG reverts to Dutch, the supported base language.

3. The IDCS domain default language, unless this language is not supported. See: Change Default Settings [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/change-default-settings.html].

   **Tip:** Use this option to set the language for a larger group of users instead of setting the preferred language for each user.

4. English (United States), the default language.

**Data Ingestion Methods and Supported Languages**

Users can ingest data into AIAMFG using either CSV files for any data or using Oracle Data Pump and Oracle GoldenGate, which is only used by E-Business Suite (EBS) customers who implement AIAMFG. Users who only ingest data using CSV files can view the AIAMFG user interface (UI) and messages in a supported language, but the languages available can change for users who ingest data using a combination of CSV files and Oracle GoldenGate or only Oracle GoldenGate. Oracle GoldenGate supports any language, which may or may not be supported by AIAMFG. This intersection of supported languages determines the languages to which the UI and messages can be translated.

**Important:** At this time, AIAMFG only translates the UI and messages, not data. You can ingest data in any language. For example, if you ingest data in Korean, the data remains stored as Korean text for all users, regardless of a user's selected language.

There are four data ingestion scenarios:

1. Only ingest data using CSV files; never enable Oracle GoldenGate. AIAMFG renders in any of the supported languages.

2. Initially ingest data using CSV files, but enable Oracle GoldenGate later. Initially, AIAMFG renders in any of the supported languages. Later, if Oracle GoldenGate ingests data, it collects data from the EBS source in any language. For example, if Oracle GoldenGate ingests data in Dutch, French, German, and Polish, the AIAMFG UI and messages only render in Dutch, French, and German, which are the common languages at the intersection between the Oracle GoldenGate data from EBS and the AIMFG supported languages. If Oracle GoldenGate ingests Spanish data later, then AIAMFG renders in Spanish, in addition to Dutch, French, and German.

3. Only ingest data using Oracle GoldenGate. Oracle GoldenGate can ingest data in any language. If any ingested languages intersect with the AIAMFG-supported languages, AIAMFG renders in those intersection languages. For example, if Oracle

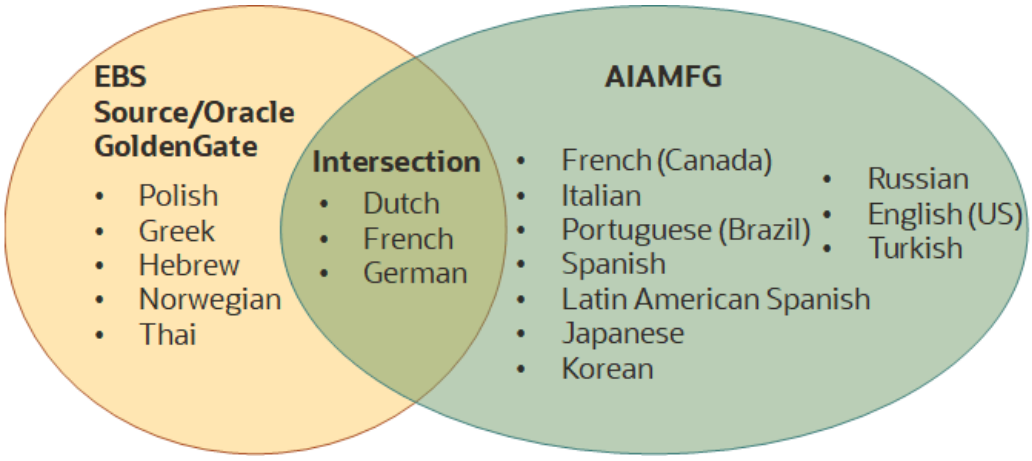
GoldenGate ingests data in Dutch, French, German, and Polish, AIAMFG renders in Dutch, French, and German.

4. Initially ingest data using Oracle GoldenGate, but ingest data using CSV files later. Initially, if any ingested languages intersect with the AIAMFG-supported languages, AIAMFG renders in those intersection languages. For example, if Oracle GoldenGate ingests data in Dutch, French, German, and Polish, AIAMFG renders in Dutch, French, and German. Later, when French, German, and Spanish data is ingested using CSV files, AIAMFG only renders in French and German. If Oracle GoldenGate ingests data in Spanish later, then AIAMFG can render in Spanish, French, and German from that point on.

   **Action:** Run the background process "Maintain Multi-lingual Tables" if you enable a supported language after transaction data already exists in other languages. The program adds the newly enabled language. When a user logs in with the new language, they will be able to view the updated data. See: Running Background Processes., page E-1

The graphic below illustrates how AIAMFG can only render in the languages at the intersection of the Oracle GoldenGate-ingested languages and the AIAMFG-supported languages. This restriction only applies when ingesting EBS data using Oracle GoldenGate.

***Supported Languages Depend Upon the Data Ingestion Method Used***




### To set up or change your preferred language:

You can navigate from AIAMFG to IDCS by selecting **Profile** from the User Name drop-down list. From the IDCS My Profile console, you can configure many of your user settings and change your password, in addition to selecting your preferred language. For information about configuring each of the user settings available from the My Profile console, see: Configure User Settings [https://docs.oracle.

com/en/cloud/paas/identity-cloud/usids/configure-user-settings1.html].

1. Navigate to the My Profile console.

   From the **User Name** drop-down list, select **Profile**.

2. In the Preferred Language field, select one of the languages supported by AIAMFG.

3. Click Save.

4. Sign out of AIAMFG, then sign in.

   The Home page now displays in your preferred language.

   > **Note:** If you selected a language not supported by AIAMFG, the application renders in the default language of English (United States).

## Defining User Preferences

The information in the Insights, Genealogy and Trace, Predictions, and Factory Command Center modules is shown in the context of the specific plant and department selected in the User Preferences page. You can set user preferences to specify how information is displayed in the module pages available for analyzing insights, genealogy and trace, predictions, and factory current status.

1. Navigate to the User Preferences page.

   Click the user name, and then click **Preferences**.

2. In the General region, specify values in the following fields. The Insights, Genealogy and Trace, Predictions, and Factory Command Center modules automatically display data only associated with the selected values.

   - **Organization Code** – Mandatory. Select a preferred organization from the list of organizations assigned to you in the Organization Access page.
   - **Category Set** – Mandatory.
   - **Department Code** – Optional, for discrete manufacturing organizations only.

3. In the Factory Command Center: Material region, optionally specify a numerical value in the following fields:

   - **Expiration Time Fence (days)** – number of days prior to the expiration date that users are notified through the Item Expiring KPI.
   - **Receipt Delay Tolerance (hours)** – number of hours beyond the receipt due

date before a receipt delay is reported.

- **WIP Return Time Fence (hours)** – number of past hours to consider and display component return transactions for open work orders.

4. In the Factory Command Center: Machine region, optionally select a value in the **Alarms last N hours** field to show the alarms during the last N hours. You can choose to view alarms up to a maximum of 24 hours.

5. In the Factory Command Center: Management region, optionally select a numerical value in the following fields:
   - **Pending Work Order Time Fence (days)** – the interval during which a soon to be started work order displays.
   - **Delay Tolerance (hours)** – the number of hours beyond which a work order or operation displays a delay in starting or completing.
   - **Completed Work Order Time Fence (days)** – the interval during which recently completed work orders display.

6. In the Logging region:
   - click the **Enable Log** button to enable logging.
   - select from the following in the **Log Level** field:
     - Error
     - Warning
     - Info
     - Statement.

7. In the Background Process Tracing region:
   - click the **Enable Trace** button to enable background process tracing.
   - select from the following in the **Trace Level** field:
     - Regular Trace
     - Trace with Binds
     - Trace with Waits
     - Trace with Binds and Waits

- select the background process to trace in the Program Name field.

8. Click **Save** to save your preferences.

# 3

# Acquiring Data

This chapter covers the following topics:

- Overview of Acquiring Data
- Overview of Data Ingestion Methods

## Overview of Acquiring Data

To analyze process manufacturing and discrete manufacturing shop floor data, you must acquire data from various data sources such as ERP applications, Manufacturing Execution System (MES), Quality/Laboratory Information Management System (LIMS) as well as from shop floor sensor device enabled machines. The solution provides adaptors to collect data from many sources. Oracle partners and customers can extract structured data from external source systems and semi structured data from machines and equipment sensor devices and load them into the data lake in Oracle Cloud. Equipment and sensor device data is contextualized with equipment and work order information and summarized for analysis.

### Overview of Acquiring Data

You can use out of the box integrations between Oracle Adaptive Intelligent Apps for Manufacturing and Oracle applications, such as Oracle E-Business Suite applications. These out of the box adapters are provided with Oracle Adaptive Intelligent Apps for Manufacturing and enable it to collect data from Oracle E-Business Suite applications. You can use REST services in Oracle Adaptive Intelligent Apps for Manufacturing to extract and load data from equipment, sensor devices, and other non-Oracle applications such as MES, LIMS or any custom data sources to Oracle Cloud. Data from these various Information Technology (IT) and Operational Technology (OT) systems can be uploaded for any historical period of time and for one or many products using periodic and incremental batch modes.

Structured data can be extracted from external source systems and loaded into the Oracle Adaptive Intelligent Apps for Manufacturing data lake in Oracle Cloud. The two types of data uploads that are supported are:

- Case Record Data
- Business Entity Data

Semi-structured data that can be acquired from shop floor sensor devices include:

- Stream Data
- Alert Data

### Acquiring Enterprise Resource Planning Data

Oracle Adaptive Intelligent Apps for Manufacturing provides adapters that enable integration with Oracle applications whose data is stored in an Oracle database. These adapters use Data Pump Import for the initial load and use Oracle GoldenGate Cloud Service for the incremental loads (see: *Oracle Cloud Using Oracle GoldenGate Cloud Service* ) to replicate data from their source systems into the database used by Oracle Adaptive Intelligent Apps for Manufacturing. The following key entities are replicated from their source systems and used for Oracle Adaptive Intelligent Apps for Manufacturing model building and analysis:

- Master data, such as items, categories, resources, resource instances, operations, routings, bills of materials, and recipes.
- Reference data, such as work orders or batches, sales orders, purchase orders, and quality test specifications.
- Transactional data, such as work order material and resource transactions, quality data, material move transactions, and work order completions.

### Acquiring Data from Other Data Sources

Oracle Adaptive Intelligent Apps for Manufacturing uses comprehensive REST services to acquire data from non-Oracle applications. REST services extract data from other data sources, such as custom or third party Enterprise Resource Planning (ERP) applications, Manufacturing Execution Systems (MES), Quality and Laboratory Information Management System (LIMS) systems, and legacy systems, as well as from machines, equipment, and sensors. Oracle partners can help users to configure and map the data from other data sources into Oracle Adaptive Intelligent Apps for Manufacturing. The REST services load the extracted data into the data lake in the Oracle Cloud.

## Overview of Importing Case Record Data

Case Record Data files capture historical work orders or assembly serial data. This upload of prepared data using the provided CSV templates can be used for quick analysis. The data is mined and prepared into flattened records and then imported into Oracle Adaptive Intelligent Apps for Manufacturing to obtain insights.

See: Importing Case Record Data, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

## Overview of Importing Business Entity Data

Business Entity Data files capture the key entities from external source systems. This detailed upload of business entities enables leveraging all AIMFG features including Insights, Predictions, Genealogy and Trace, and Factory Command Center. The Business Entity file upload captures data for the individual business entities like items, lots, departments, person, machines, receiving, work orders, quality, and so on and Oracle Adaptive Intelligent Apps for Manufacturing processes all the uploaded data together for analysis by understanding the underlying data structures.

Business Entity Data can be imported using the provided CSV templates, or REST services, or out of the box integrations like Oracle Data Pump or GoldenGate (for E-Business Suite only).

See: Importing Business Entity Data, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

## Overview of Importing Sensor Devices Data

Sensor Stream data (like temperature) and Alert data (like Machine Status such as up, down, idle, or in-use) from shop floor sensor devices can be imported into Oracle Adaptive Intelligent Apps for Manufacturing using CSV files. The sensor data is contextualized with the Business Entity data and summarized for analysis. Separate CSV templates are provided to capture sensor stream and alert data.

Before you can use a sensor device to acquire and process data, use the Create Sensor Device Mappings page to:

- Enter the sensor device ID and attribute details such as manufacturer, model number, and serial number.
- Map the defined sensor device to an ERP equipment and ERP equipment instance.
- Map each sensor device stream parameter to an ERP equipment parameter and parameter Unit of Measure (UOM).
- Set the stream sample interval for processing events.
- Select a time series feature set name in production analysis usage for extracting features for model analysis.
- Select a times series feature set name in machine event analysis usage for event identification process.
- Specify the rules for equipment and work order contextualization.

**Overview of Time Series Data:**

Time Series data from the equipment sensor devices comprises of a sequence of values or events obtained over a period of time and recorded at specific intervals. You can define time series feature sets to extract features for model analysis or detecting events from time series sensor device stream data.

You can create time-series feature sets by selecting a specific time segment and a specific simple function, for example like standard deviation, or a Symbolic Aggregate approXimation advanced function. Symbolic Aggregate approXimation (SAX) is a symbolic representation for complex time series data.

See:

- Defining Time Series Features Sets, page 5-2
- Setting Up Time Series Features Sets, page 5-8

Oracle partners can help users to configure machine data acquisition systems, such as Supervisory Control and Data Acquisition (SCADA), Distributed Control Systems (DCS) and other gateway device systems, to extract machine sensor data in the CSV file format. You can then upload these sensor data files into Oracle Adaptive Intelligent Apps for Manufacturing in batch mode. The application processes the sensor stream data and alert data, contextualizes it with equipment and work order information, and summarizes the contextualized data for analysis.

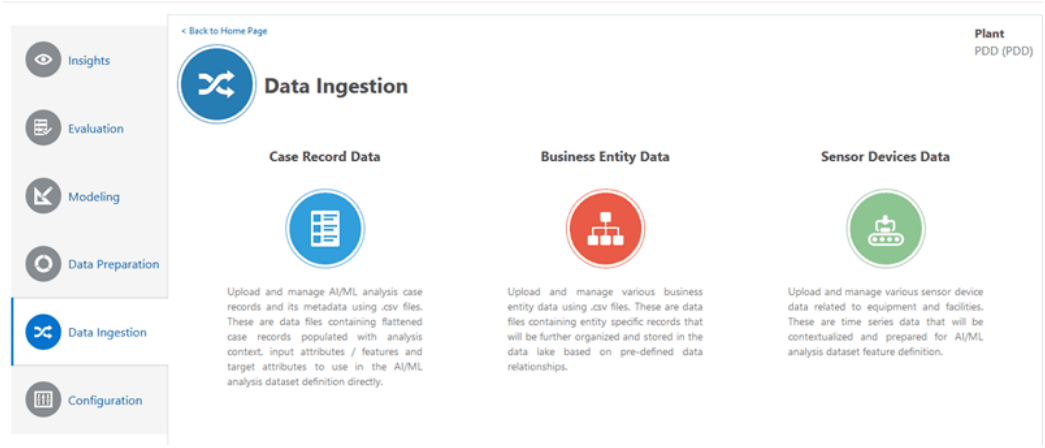
See: Importing Sensor Devices Data, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

## Overview of Data Ingestion Methods

The following data ingestion methods are available to ingest data into Oracle Adaptive Intelligent Apps for Manufacturing:

- Comma separated values (CSV) files can be used to import both structured and semi-structured data.
- REST Services are provided to import data into Oracle Adaptive Intelligent Apps for Manufacturing using CSV files.
- Oracle Data Pump and GoldenGate (for Oracle E-Business Suite only) are out of the box integration provided between E-Business Suite and Oracle Adaptive Intelligent Apps for Manufacturing enabling high-speed transfer of data and metadata from source database to target database.

Oracle Adaptive Intelligent Apps for Manufacturing provides specific CSV templates, and Data Ingestion user interfaces to upload Case Record Data, Business Entity Data, and Sensor Devices Data. From the **Home** page, click **Insights** or **Predictions**, then **Data Ingestion**. You can then select the specific user interface, depending on the data you would like to upload.




Note that you must define and be assigned to a Case Record Data organization to upload case record data files and to a Business Entity Data organization to upload the business entity data files. See Setting Up Data Ingestion, *Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

See the *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide* for detailed information on uploading and importing data from various manufacturing enterprise systems and machine sensor devices into Oracle Adaptive Intelligent Apps for Manufacturing.

# 4

# Storing Data

This chapter covers the following topics:

- Overview of Storing Data
- Using the Database Cloud Service
- Using the Storage Cloud Service

## Overview of Storing Data

The robust and comprehensive data lake used by Oracle Adaptive Intelligent Apps for Manufacturing contains the following three types of data stores:

- Relational/structured data store to store data from various relational data sources such as Enterprise Resource Planning (ERP), Supply Chain Management (SCM), Customer Relationship Management (CRM), Human Capital Management (HCM), Manufacturing Execution Systems (MES), and Quality systems.
- Time series data store to store data from various machines and sensors such as temperature, pressure, and vibration as well as from external environmental sources such as temperature and humidity.
- Unstructured data store to store information such as notes, comments, and log files.

Oracle Adaptive Intelligent Apps for Manufacturing stores manufacturing operations data in the following two databases:

- Oracle Database Cloud Service
- Oracle Storage Cloud Service

Oracle Database Cloud Service and Oracle Storage Cloud Service are underlying architectural components of Oracle Adaptive Intelligent Apps for Manufacturing. General information about how Oracle Adaptive Intelligent Apps for Manufacturing uses each database is provided below. For detailed information about these two

databases, see:

- Oracle Database Cloud Service [https://docs.oracle.com/en/cloud/paas/database-dbaas-cloud/index.html]
- Oracle Cloud Infrastructure Object Storage Documentation [https://docs.us-phoenix-1.oraclecloud.com/Content/Object/Concepts/objectstorageoverview.htm]

## Using the Database Cloud Service

Oracle Database Cloud Service serves as the data lake for Oracle Adaptive Intelligent Apps for Manufacturing by storing structured data used for analysis. The types of data stored in this database include Enterprise Resource Planning (ERP) application data, Manufacturing Execution System (MES) data, and Quality/LIMS, sensor device mapping definition, and summarized machine sensor data. This data is used for model building and analysis, lot genealogy, presenting the manufacturing time line, and providing a real-time overview of factory events in the Factory Command Center.

## Using the Storage Cloud Service

Oracle Storage Cloud Service serves as the data lake for semi-structured data. The types of data stored in this database include raw sensor data and contextualized machine sensor data in parquet partition tables. It stores the granular data of both stream and alert data processed from the sensor data files. Save the sensor data in CSV file format and store it in the Storage Cloud. Specify the file path in the Upload Files page in the Data tab of the Sensor Devices page. Storage cloud also stores the sensor device mapping definition CSV files which are used to import sensor device mappings.

# 5

# Preparing Data

This chapter covers the following topics:

- Overview of Preparing Data
- Understanding Time Series Feature Sets
- Setting Up Time Series Feature Sets
- Defining Key Performance Indicators
- Creating Datasets for Analysis
- Viewing Sensor Summary Results

## Overview of Preparing Data

Before creating data models, you must create datasets for analysis by extracting the features from time series data and from structured data.

Use time series feature sets to define the slicing of time series data into multiple time segments and summarize the data using various functions such as average, standard deviation, min, max, and so on.

Use the dataset you create to prepare features from structured entities of the data lake such as operation duration, resource usage, operator worked, and so on, that are grouped under the Manpower, Machine, Material, Method and Management categories. Datasets also prepare features from the time series data based on the summary functions defined in the time series feature sets.

Use sensor summary results to review the summarized values of the time series features and compare the values across the work orders or serial units. The solution provides configuration of custom specific features using web services, which can be used for the analysis.

## Understanding Time Series Feature Sets

Time series data consists of a sequence of values or events obtained over a period of time. It is an ordered data set with data points in specified intervals. Time series data are very high dimensional, noisy and covariant. This makes the data difficult to use for basic statistical operations or other complex data mining tasks. As it is challenging to directly use time series data in machine learning algorithms, time series feature sets definitions provide a flexible and extensible way to define and compute summaries from time series data. These time series features can then be used for model building and analysis.

Time-series feature sets can be used for:

- Production Analysis Usage - by extracting features from the sensor stream data for model building. On creating a dataset, contextualized stream data from the sensor devices is used to generate the sensor summary data in the duration of actual work order operation.

  **Note:** While the machine can send sensor data when the machine is in faulted or in an idle state, for analysis purposes only the sensor data available during the actual operation execution is used. Any data outside the actual operation execution is not considered for model analysis.

- Machine Event Analysis Usage- by deriving alerts from the sensor stream data based on threshold violation or SAX pattern match rules. It is used only for event identification and is not used for model analysis.

Time-series feature sets are created from the combination of the specific time segment and the specific simple function or advanced function you choose:

- **Time Segments**:

  Segmentation is the process of dividing the time series into segments or subsets. Each time segment is treated independently and a specific segment can be chosen/used for model building.

  For example, an ERP equipment instance works on a particular work order or batch for a certain duration, then remains idle for the rest of the time. Sensor data from a manufacturing process can be functionally segmented based on the work order information. You can choose the time segment for viewing sensor summary. You can choose the required time segment during model building.

  The following are the three options to define time segments:

  - Fixed: Divides time series into segments of fixed duration.

    For example, for a work order of 5 hours duration, assuming that each segment

is fixed for one hour, the time series data is divided into five segments each of one hour duration.

- Sliding: Divides time series into segments of the same duration over a sliding interval.

  For example, for a work order of 5 hours duration, if you apply a sliding segment of one hour and a sliding interval of 10 minutes, the time series data is divided into segments each of one hour and segmented over 10 minutes interval. In this example, it will generate 25 time segments in minutes from 0 to 60, 10 to 70, 20 to 80 and so on, up to 240 to 300 minutes.

- Full: Sets a single segment over the entire duration of equipment usage.

  For example, for a work order of 5 hours duration, a full segment covers the entire duration of work order operation which is 300 minutes.

  **Note:** Duration and interval unit is in minutes.

- **Simple Functions**

  Time series data in each time segment can be further transformed before computing the summary or events.

  For production analysis, the available simple functions are:

  - Average: Average of all data points
  - Standard Deviation: Standard deviation of all data points
  - Minimum: Minimum value of all data points
  - Maximum: Maximum value of all data points
  - Count Above Threshold: Number of data points above the threshold value
  - Count Below Threshold: Number of data points below the threshold value
  - Count Within Range: Number of data points inside a range value specified
  - Count Outside Range: Number of data points outside a range value specified

  For machine event analysis, the available simple functions are:

  - Above Threshold Alert: Alert event when a data point goes above the threshold value
  - Below Threshold Alert: Alert event when a data point goes below the threshold value

- Within Range Alert: Alert event when a data point falls inside a range value specified
- Outside Range Alert: Alert event when a data point falls outside a range value specified

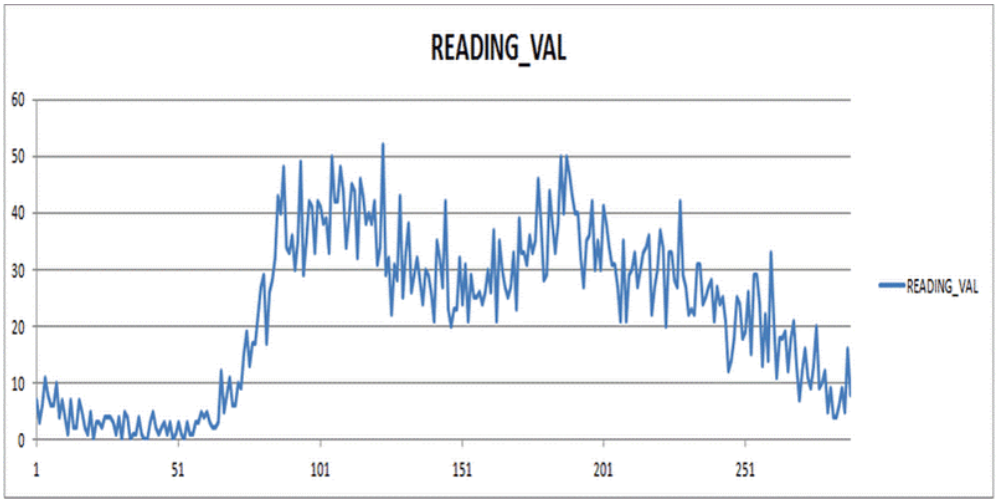
- **Advanced Functions**

  The time-series feature set definition can make use of Symbolic Aggregate approXimation advanced functions. Symbolic Aggregate approXimation (SAX) is a symbolic representation for complex time series data.

  For example, temperature readings from a furnace every millisecond can have 86400000 records in a day. In order to perform any efficient data mining on such massive datasets, the time series data is transformed into another form which retains most of the information in the original raw data. Data mining can then be performed on such approximated transformation. Symbolic Aggregate approXimation creates an approximation of this data which fits in the main memory, but retains the essential features of interest. It converts the time-series numeric values to symbolic text.

  **Using SAX with Time Series Data**:

  The following graph (where in the X-axis represents time and the Y-axis represents reading values) shows time series data for which readings have been taken every five minutes bringing it to 288 data points per day:

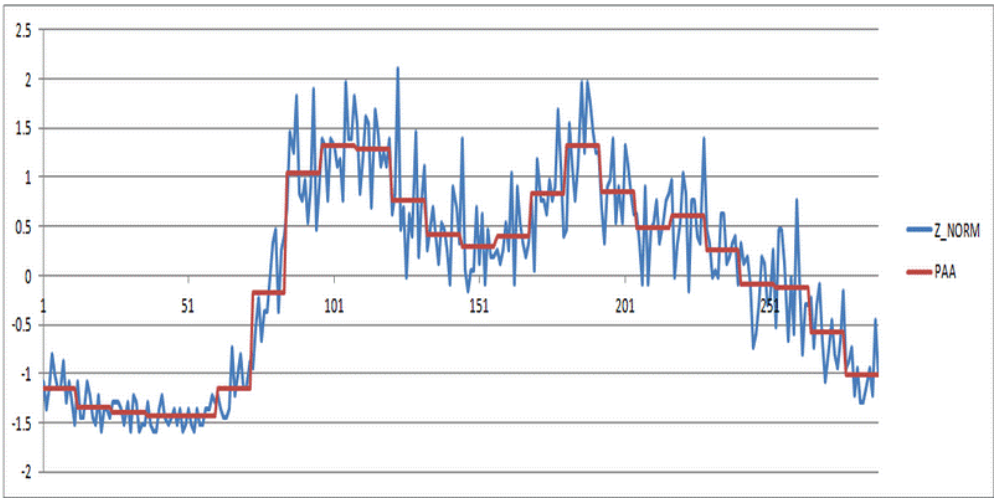



  SAX is computed for time series data by first applying Z-normalization which converts the time series data with the mean value of zero. This standardizes the data to enable comparing shapes and patterns without losing the information in the original raw data.

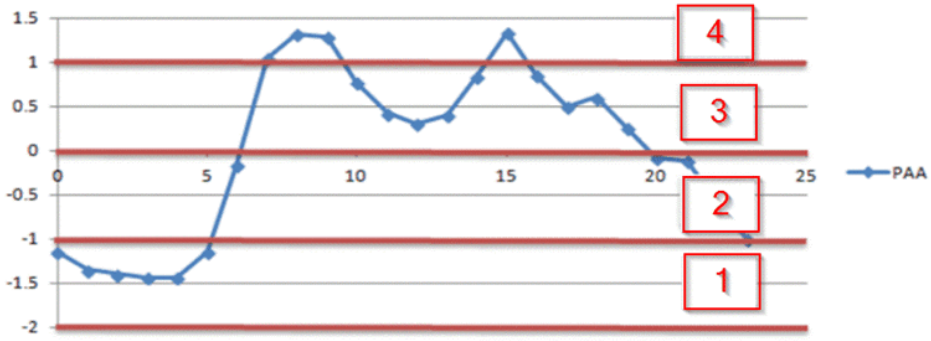
  The Piecewise Aggregate Approximation (PAA) of the time series is then calculated using the sample interval value and aggregating the points without losing the

shape of the time series.

The following graph shows the time series data after Z-normalization and PAA:




The Y-axis is then divided into bands based on the alphabet size assuming the normal distribution of data across the Y-axis. The following graph shows four bands on the PAA of the time series:




The time series after the computation of SAX is converted to the following symbolic text: 11111244433333433332222

This symbolic representation can be used for various data mining techniques.

For production analysis time series feature sets, the available advanced functions are:

- SAX Bitmap Count: Depending on the SAX Bitmap size, the time series sequence is checked for the number of occurrences of the SAX Bitmap combination.

- SAX Pattern Count: Depending on the user specified SAX pattern the time series sequence is checked for the number of occurrences of the specified SAX

matching pattern. Users can specify patterns as regular expressions like, for example, 1234, 12, 132, and so on based on SAX alphabet size.

For machine event analysis time series feature sets, the available advanced function is:

- SAX Pattern Alert: Depending on the user specified pattern, the time series sequence is checked for the specified pattern, and an alert is issued when it occurs.

The combination of each function (simple or advanced function) you select when you create a time-series features set becomes a feature or an event. For example, within the boundary of a step, if you have 3 fixed time segments, you will derive 3 features and each feature will be computed using the function. If the function selected was Average, then there will be 3 features as segment1-average, segment2-average, and segment3-average. These features, generated by time-series feature sets, are available as features during model building.

For example, if you give the time segment as Full, SAX Alphabet Size as 4, and the SAX Bitmap Size as 1, the following four features are extracted:

- Full – SAX Bitmap (1)
- Full – SAX Bitmap (2)
- Full – SAX Bitmap (3)
- Full – SAX Bitmap (4)

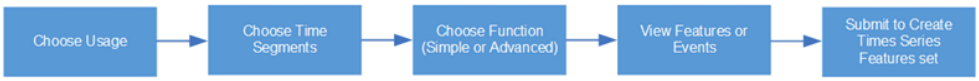
Similarly, if you give the time segment as Full, SAX Alphabet Size as 4, and the SAX Bitmap Size as 2, the following 16 features are extracted:

- Full – SAX Bitmap (11)
- Full – SAX Bitmap (12)
- Full – SAX Bitmap (13)
- Full – SAX Bitmap (14)
- Full – SAX Bitmap (21)
- Full – SAX Bitmap (22)
- Full – SAX Bitmap (23)
- Full – SAX Bitmap (24)
- Full – SAX Bitmap (31)

- Full – SAX Bitmap (32)
- Full – SAX Bitmap (33)
- Full – SAX Bitmap (34)
- Full – SAX Bitmap (41)
- Full – SAX Bitmap (42)
- Full – SAX Bitmap (43)
- Full – SAX Bitmap (44)

Basically, patterns in the time series data are matched with the SAX bitmap pattern (shown in brackets like 11, 12, and so on) and it can find the number of times it is matched in the time series data.

## Process Flow

1. 


   The process of setting up a time series feature set begins by choosing what you will use it for. You can use a time series feature set for:

   - Production Analysis: For data mining purposes, the features are extracted and used while building models for insights and predictions.
   - Machine Event Analysis: While alert information is directly obtained from the sensor devices, it can also be derived from time series sensor stream data by running the event identification processors.

2. Select the Time Segment which basically, within the boundary of a work order, operation, or step, divides the time series data in to fixed segments, sliding segments, and full time segments.

   **Note:** For Machine Alert Analysis usage you can only select full time segments.

3. You can choose to select from the options available for either from Simple or Advanced functions.

4. View the features or events created from the combination of each time segment and each function. In data mining, features generated are used in model building. Note that events are used for deriving alerts from sensor stream data using the event

identification processors.

5. Submit a Time Series Feature set. Later, you can select the set for use when setting up sensor devices.

See: Setting Up Time Series Feature Sets, page 5-8

## Setting Up Time Series Feature Sets

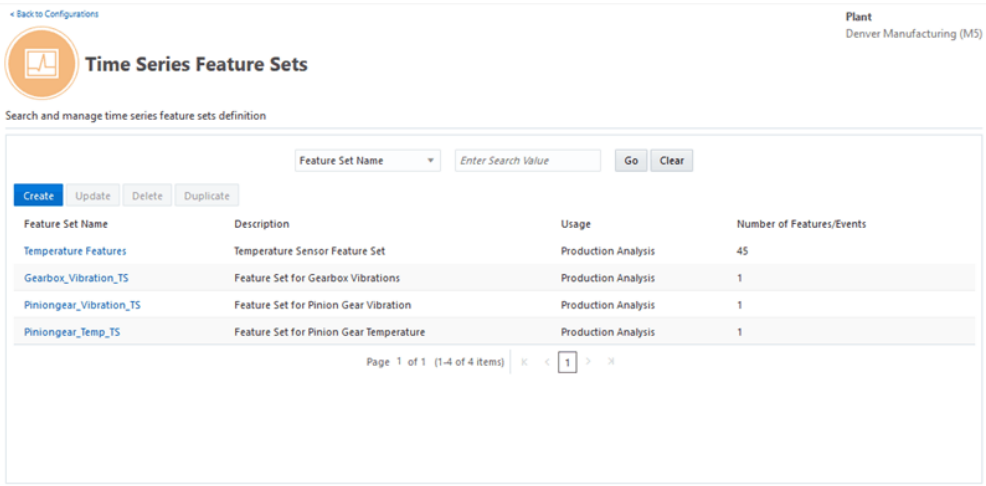
Use the Time Series Feature Sets page to:

- View existing time series feature sets.
- Create a new time series feature set.
- Update an existing time series feature set.
- Delete an existing time series feature set.
- Duplicate an existing time series feature set.

### To view a time series feature set:

1. Navigate to the Time Series Feature Sets page.

   From the Home Page, click **Insights** or **Predictions**. Click the **Configuration** link, then **Time Series Feature Sets**.




2. The existing time series feature sets display in the search results table in the Time Series Features Set page. Columns include:
   - Feature set name

- Description
- Feature set usage
- Number of features/events

3. To view details of a specific feature set definition, click on a Feature Set Name value in the search results table. The View Time Series Feature Set page appears with the details of the time series feature set you selected.


You can view the sample data chart and the time segment and function applied on the chart.

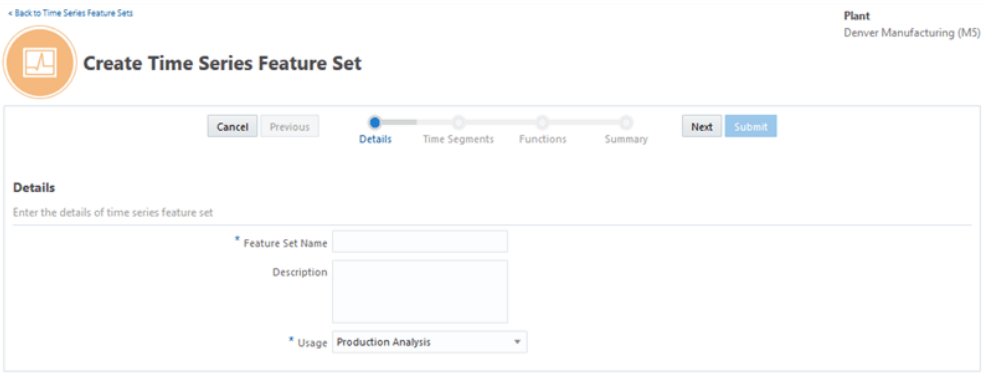
4. Click **Configure Sample** to select the range, interval and number of data points to be plotted on the sample chart. You can then view the raw chart and the value of the function based on the data points plotted on the chart. When the selected feature or event has SAX function, it shows the raw chart and the SAX chart to understand the function that is used.

### To create a time-series feature set:

1. You can define and use time series feature sets using the Create Time Series Feature Set page. These time series feature sets definition can then be applied to time series contextualized stream data, making the data easier to use for analysis.

   Navigate to the Create Time Series Feature Set page.

2. In the Time Series Features Set page, click **Create**.

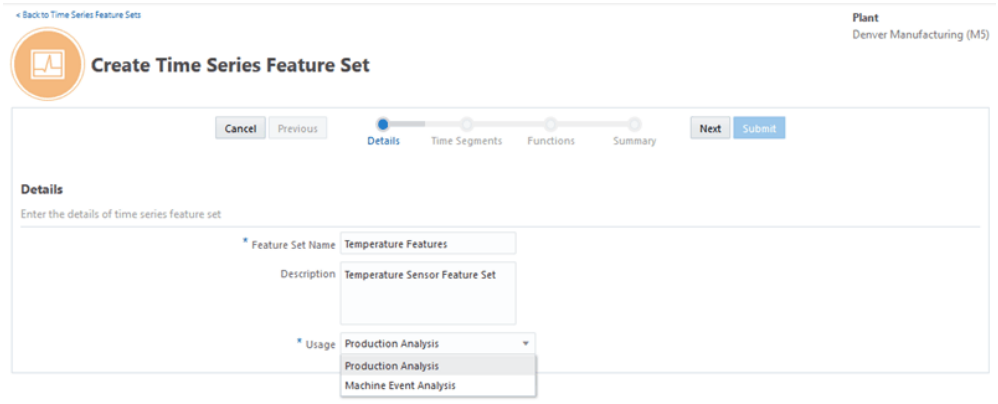



3. Use the Create Time Series Feature Set page to enter the details, time segments, functions and view summary. Enter:

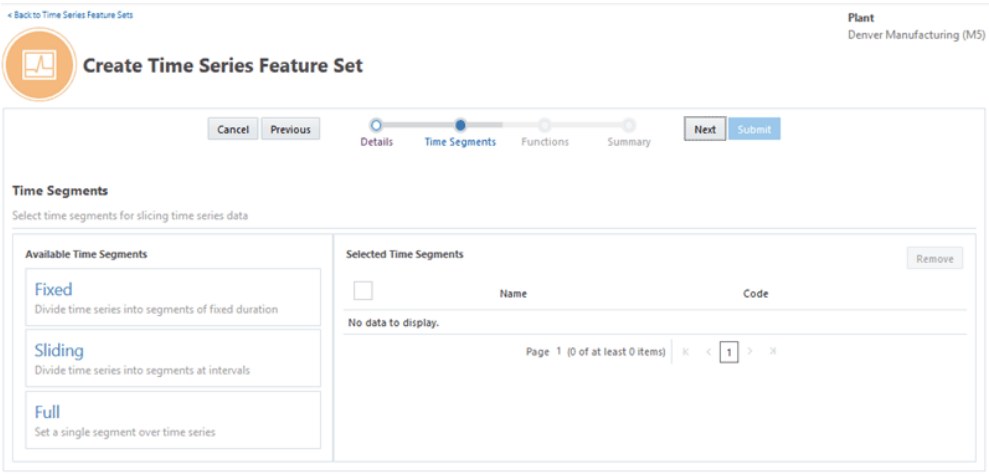
   - **Feature Set Name**
   - **Description**

4. In the **Usage** field select from:

   - Production Analysis: To derive insights and predictions from the time series features, for data mining purposes.
   - Machine Event Analysis: To derive alerts from sensor stream data using the event identification processors.




5. Click **Next**.




6. Select the time segment in the Available Time Segments region. Depending on the parameter defined, different time segments are generated. Choose from the following Time Segments:

   - Fixed
   - Sliding
   - Full

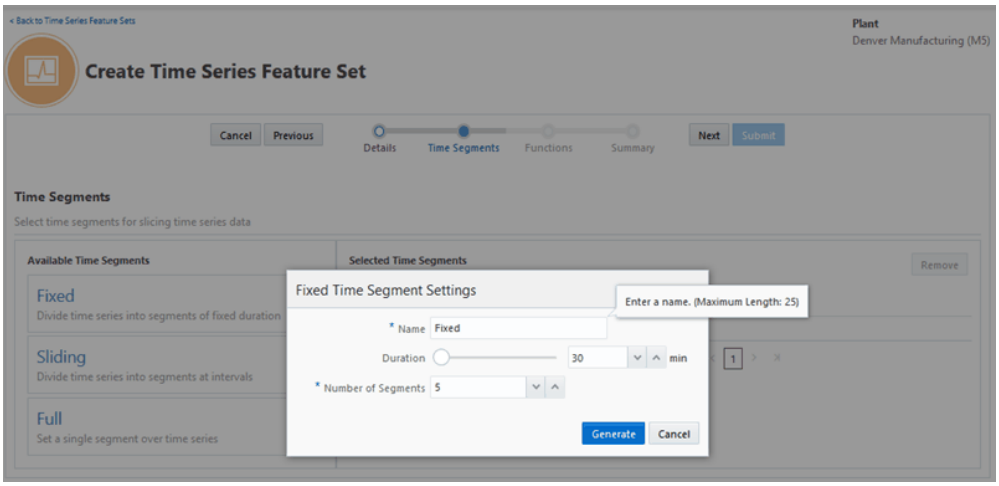
   For production analysis, select from fixed, sliding, or full segments.

   For Machine Alert Analysis you can only select full segment.

7. To select a fixed segment, click Fixed.

8. In Fixed Time Segments Settings, enter:

   - **Name**
   - **Duration** in minutes
   - **Number of Segments**

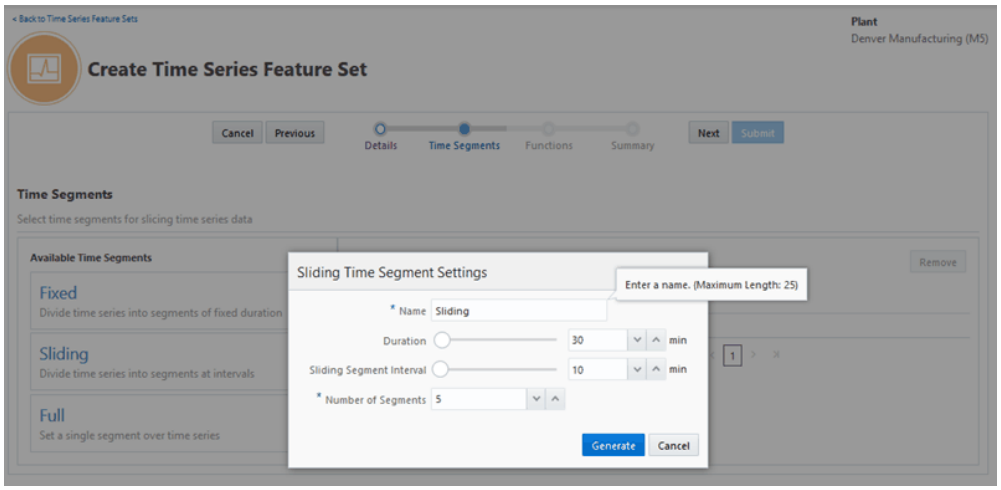



Click **Generate** to create the fixed time segment. Click **Cancel** to cancel your selection.

9. To select a sliding segment, click Sliding.

10. In Sliding Time Segments Settings, enter:

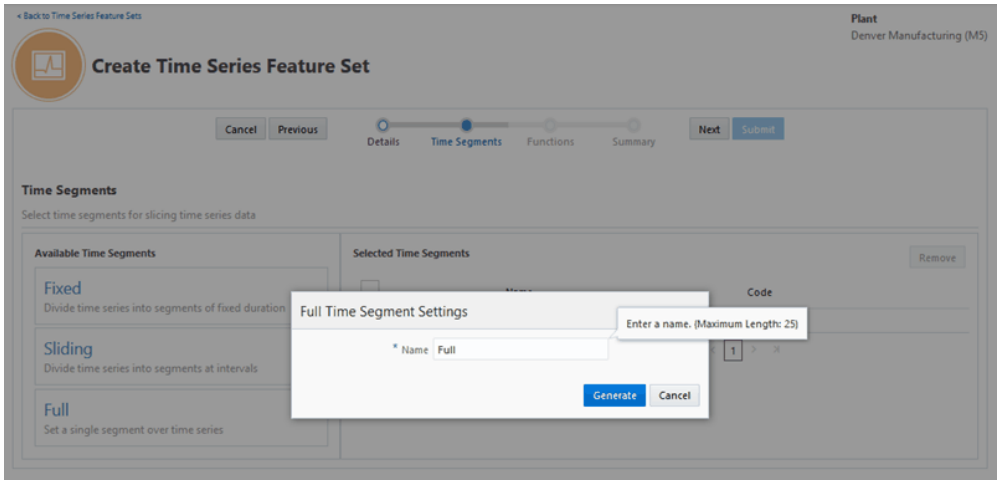
    - **Name**
    - **Duration** in minutes
    - **Sliding Segment Interval** in minutes
    - **Number of Segments**




Click **Generate** to create the sliding time segment. Click **Cancel** to cancel your selection.

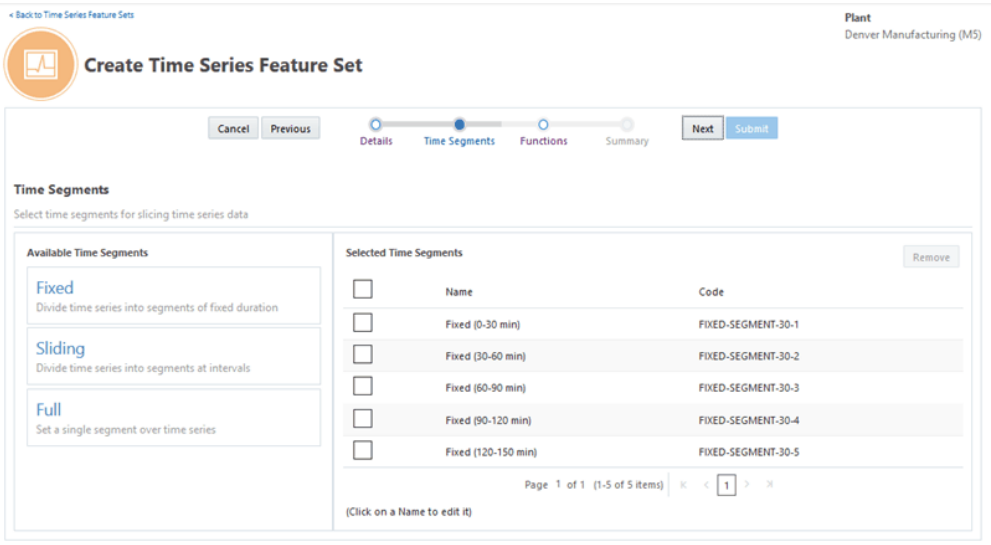
11. To select a full segment, click Full.

12. In Full Time Segment Settings, enter **Name** . There are no additional fields for full time segments.




Click **Generate** to create the full time segment. Click **Cancel** to cancel your selection.

13. The time segments you generate appear in the Selected Time Segments region.




To remove a time segment, select the check box next to the time segment and click **Remove**.

> **Note:** When you select a row in the table and select check box, all the associated time segments that were created along with the selected row are grouped and selected for removal.

14. You can select to update the time segment name only. Click the time segment name, enter your changes in settings and click **Update**.

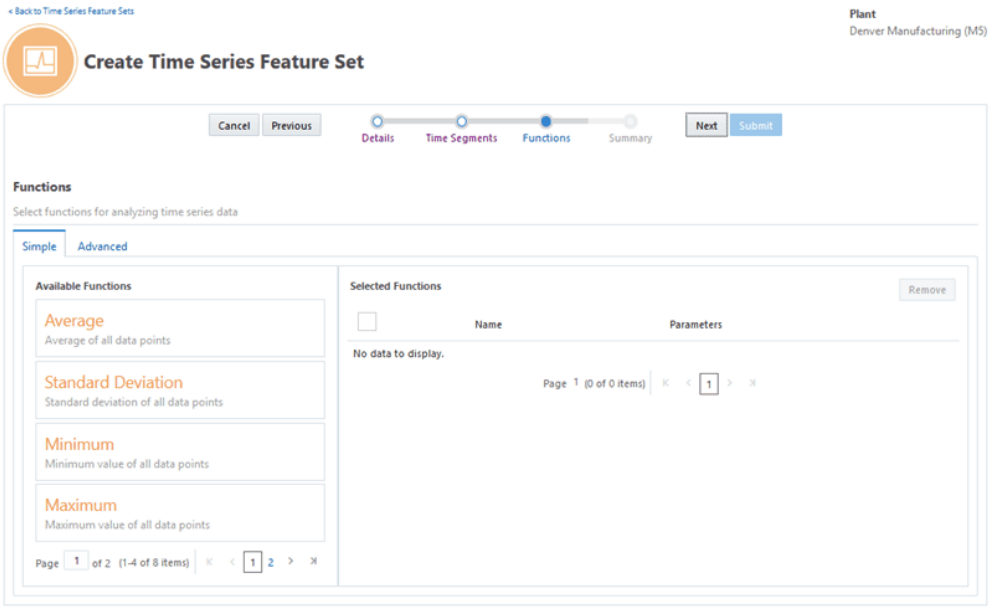
15. Click **Next**.

16. Select a simple function from the list appearing in the Simple functions tab. Enter the parameters for the simple function you select, and click **Generate**.

    For production analysis usage, click on any one of the following simple functions that appear as tiles:

    - Average
    - Standard Deviation
    - Minimum
    - Maximum
    - Count Above Threshold
    - Count Below Threshold

- Count Within Range
- Count Outside Range




For production analysis usage, if you select Average, Standard Deviation, Minimum or Maximum, enter **Name**. There are no other parameters.

For production analysis usage, if you select Count above Threshold or Count below Threshold, enter:

- **Name**
- **Threshold Value**

For production analysis usage, if you select Count within Range or Count outside Range, enter:

- **Name**
- **Range Start**
- **Range End**

For machine event analysis, select one of the following simple functions:

- Above Threshold Alert
- Below Threshold Alert
- Within Range Alert

- Outside Range Alert

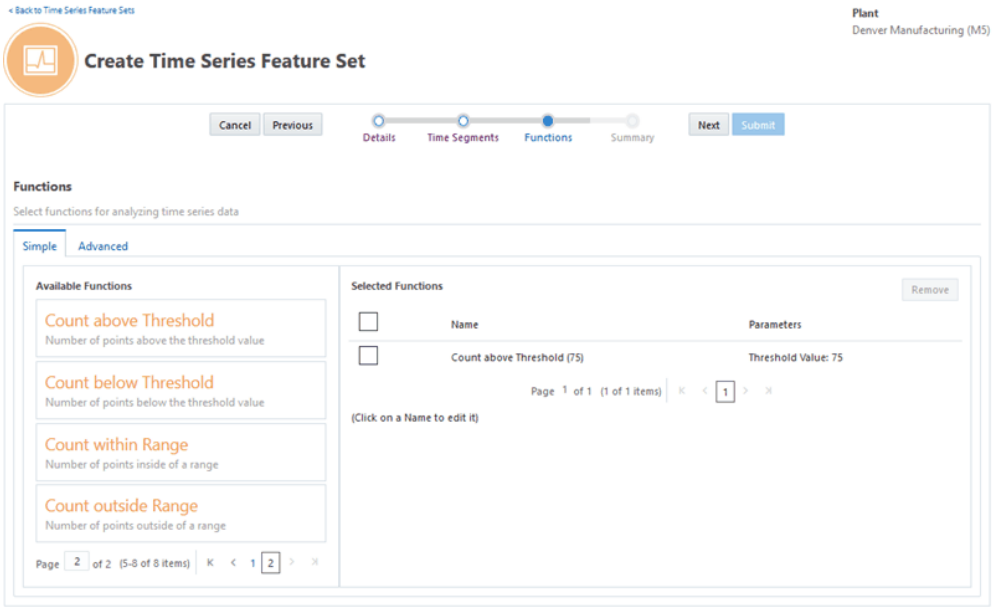
For machine event analysis usage, if you select Above Threshold Alert or Below Threshold Alert, enter:

- **Name**
- **Threshold Value**
- **After Match** - select Skip To Last or Skip to Next. The default value is Skip to Last.
- **Value Aggregation Function** - select from Average, Minimum and Maximum. The default value is Average.

For machine event analysis usage, if you select Within Range Alert or Outside Range Alert, enter

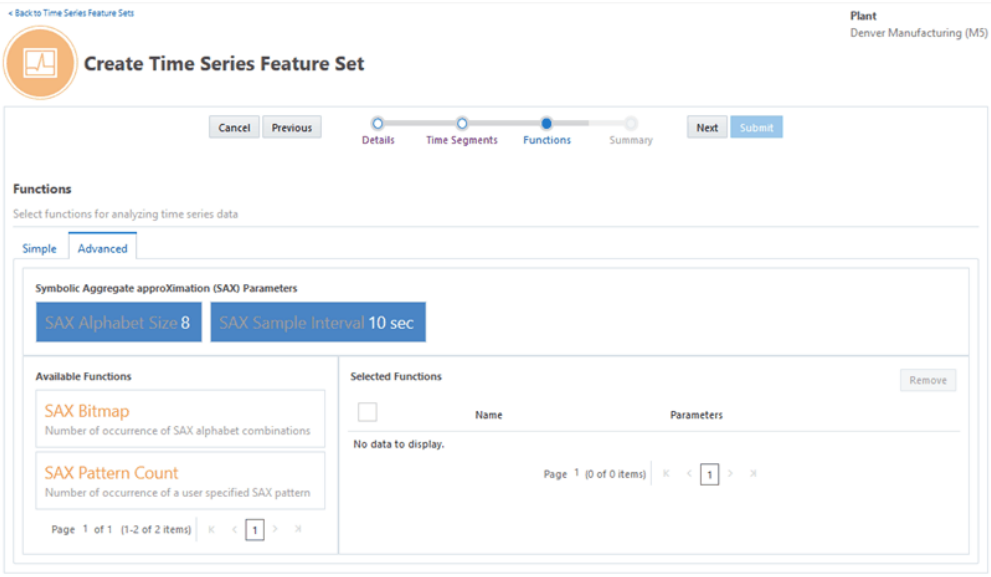
- **Name**
- **Range Start**
- **Range End**
- **After Match** - select Skip To Last or Skip to Next. The default value is Skip to Last.
- **Value Aggregation Function** - select from Average, Minimum and Maximum. The default value is Average.

17. The selected simple functions you generate appear in the Selected Functions region.




To remove a selected function, select the check box next to the simple function and click **Remove**.

18. Click the **Advanced** tab to select an advanced function.




19. In SAX Parameters, select the size or number of SAX bands in the **SAX Alphabet Size** field. The number of bands supported are 4, 6 or 8. The default value is 8.

20. Select a value in the **SAX Sample Interval** field in seconds, minutes, or hours. The value should be greater than zero. The default value is 10 sec.

21. From Available Functions, select an advanced function. For production analysis usage, select from the following list of advanced functions:

    - SAX Bitmap Count
    - SAX Pattern Count

    For machine event analysis, select the following advanced function:

    - SAX Pattern Alert

22. To select SAX Bitmap Count for production analysis, enter:

    - **Name**
    - **Bitmap Size** - select from 1 bit or 2 bits. The default value is 1 bit.

    Click **Generate** to create the SAX Bitmap Count. Click **Cancel** to cancel your selection.

23. To select SAX Pattern Count for production analysis, enter:

    - **Name**
    - **Pattern (regex)** - enter an expression for the pattern match. The default value is 1234.
    - **After Match** - select Skip to Last or Skip to Next. The default value is Skip to Last.

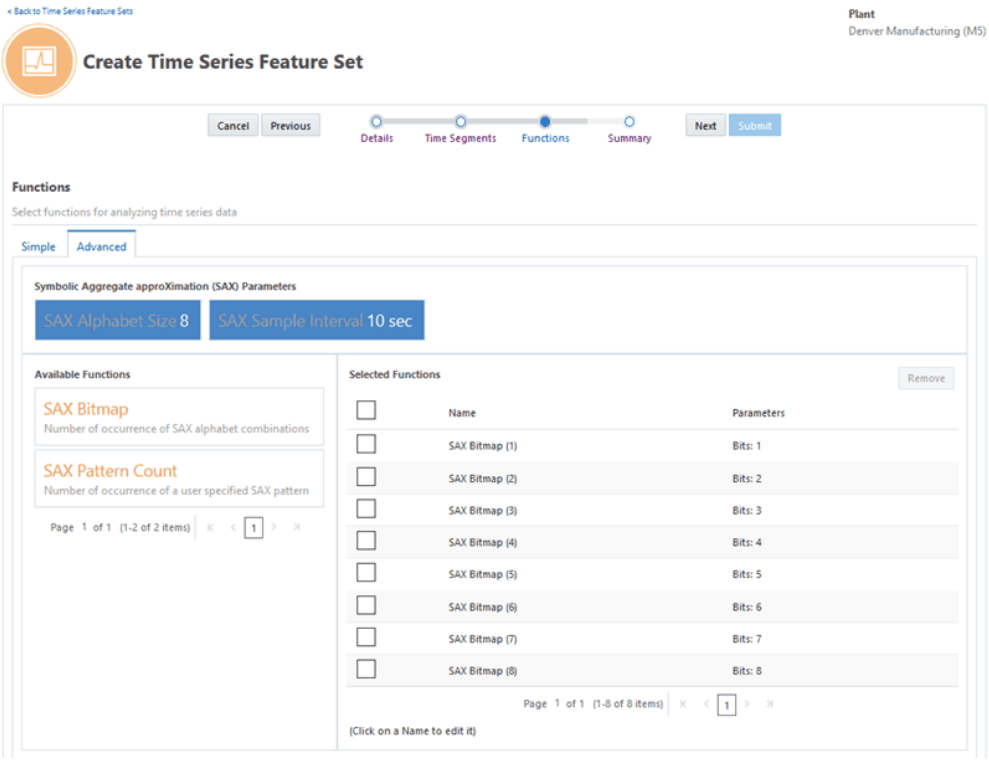
    Click **Generate** to create the SAX pattern. Click **Cancel** to cancel your selection

24. To select SAX Pattern Alert for machine event analysis, enter.

    - **Name**
    - **Pattern (regex)** - enter an expression for the pattern match. The default value is 1234.
    - **After Match** - select Skip to Last or Skip to Next. The default value is Skip to Last.
    - **Value Aggregation Function** - select from Average, Minimum, and Maximum. The default value is Average.

    Click **Generate** to create the SAX pattern alert. Click **Cancel** to cancel your selection.

25. The selected advanced functions you generate appear in the Selected Functions region.

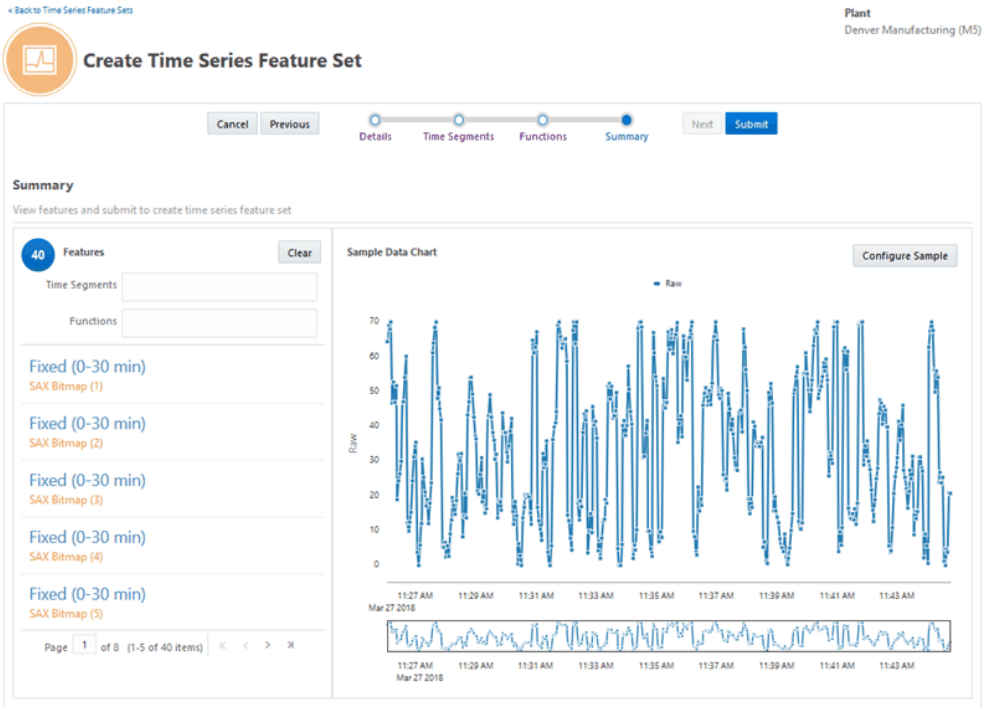



To remove an advanced function, select the check box next to the advanced function and click **Remove**.

> **Note:** When you select a check box in the selected functions table, all the functions that were created along with it are grouped and selected for removal.

26. Click **Next**.

27. In the Summary, view the created features. The combination of each time segment and each function you select becomes a feature or an event.




The Summary shows the number of features/events created from the combination of the time segment and function that you have selected. Click **Features/Events** tile to view the sample data chart.

You can filter the features/events that appears using the **Time Segments** field and/or **Functions** field.

The Sample Data Chart region shows the sample data chart and the time segment/function applied on the chart. SAX chart is displayed only when user selects the feature/event with SAX function.

28. Click **Submit**. Depending on the usage you have selected, you can now use the time series feature set you have created when creating sensor devices mappings for production analysis or machine event analysis of time series sensor stream data.

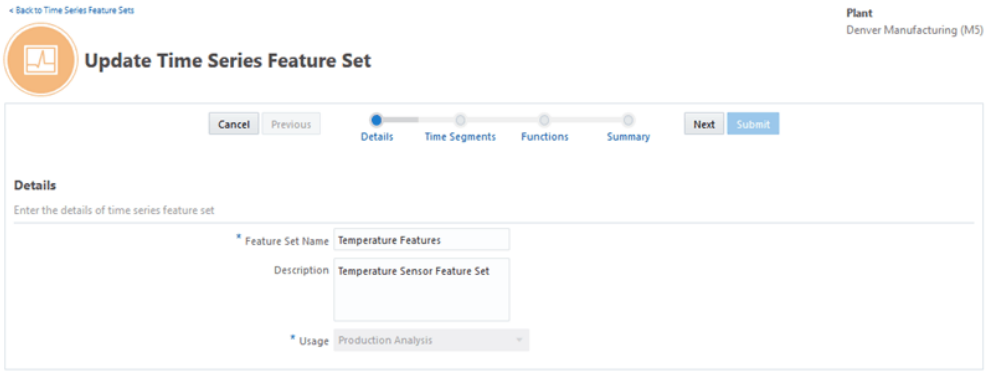
    See: Setting Up Sensor Devices Mapping, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

### To update a time series feature set:

1. To update a time series feature set, navigate to the Update Time Series Feature Set page.

   In the Time Series Feature Sets page, select the time series feature set you would like to update.

2. Click **Update**.



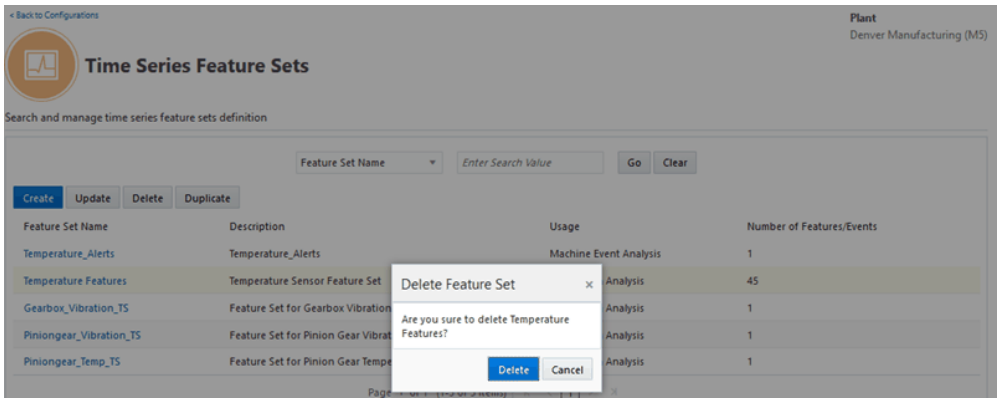

3. Update the Features Set Name and Description details of the time series feature set. Note that you cannot update the Usage field.

4. Update your selections for time segments and simple or advanced functions. View the summary information, and click **Submit**.

### To delete a time series feature set:

1. To delete a time series feature set, navigate to the Time Series Feature Sets page, select a time series feature set. .

2. Click **Delete**.



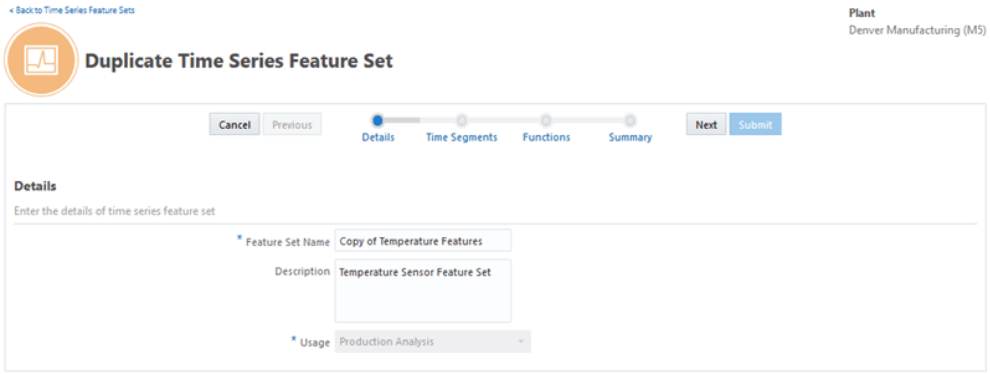

3. In Delete Feature Set notification that appears, click **Delete**. To retain the feature set, click **Cancel**.

### To duplicate a time series feature set:

1. To duplicate an existing time series feature set, navigate to the Duplicate Time Series Feature Set page.

   Select the time series feature set you would like to duplicate.

2. Click **Duplicate**.




3. Enter the Feature Set Name and Description details of the time series feature set. Note that the Usage field is copied from the time series feature set you duplicated and cannot be changed.

4. You can duplicate or change the selections for time segments and simple or advanced functions. View the summary information, and click **Submit**.

## Defining Key Performance Indicators

Key performance indicators (KPI) are required for insights and prediction analysis in both process manufacturing and discrete manufacturing organizations. You can use the Key Performance Indicators page to define and manage setups for KPIs, model target attributes, and specify target bins for machine learning analysis. Note that the KPIs you define apply to all organizations.

**Seeded Key Performance Indicators**

The application supports the following four seeded KPIs:

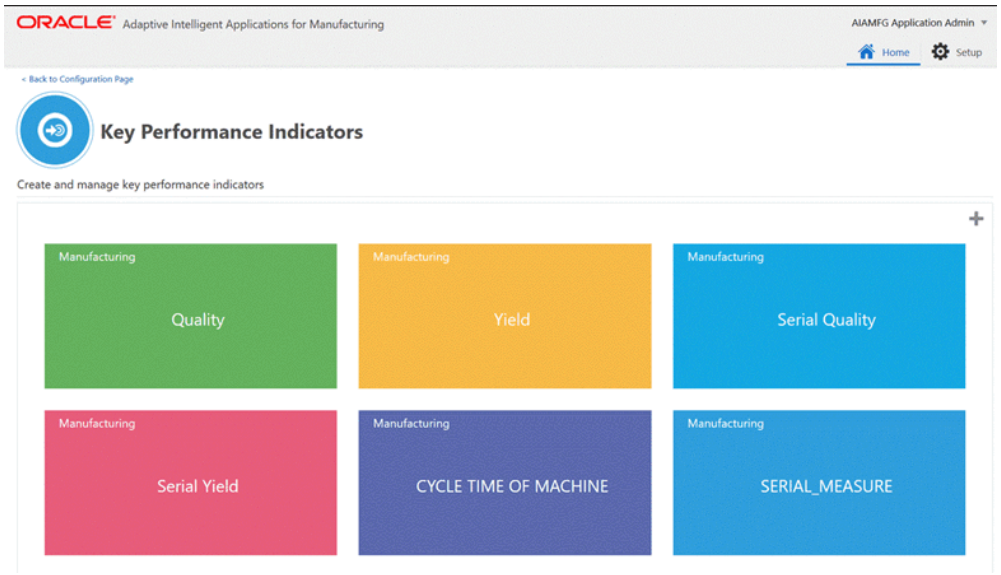
- Yield
- Quality
- Serial Unit Yield
- Serial Unit Quality

**Custom Key Performance Indicators**

You can also create custom KPIs such as Cycle Time, Machine Efficiency, Machine Downtime and so on. You can map custom KPIs to attributes, and use them for Insights and Predictions analysis.

### To define a key performance indicator:

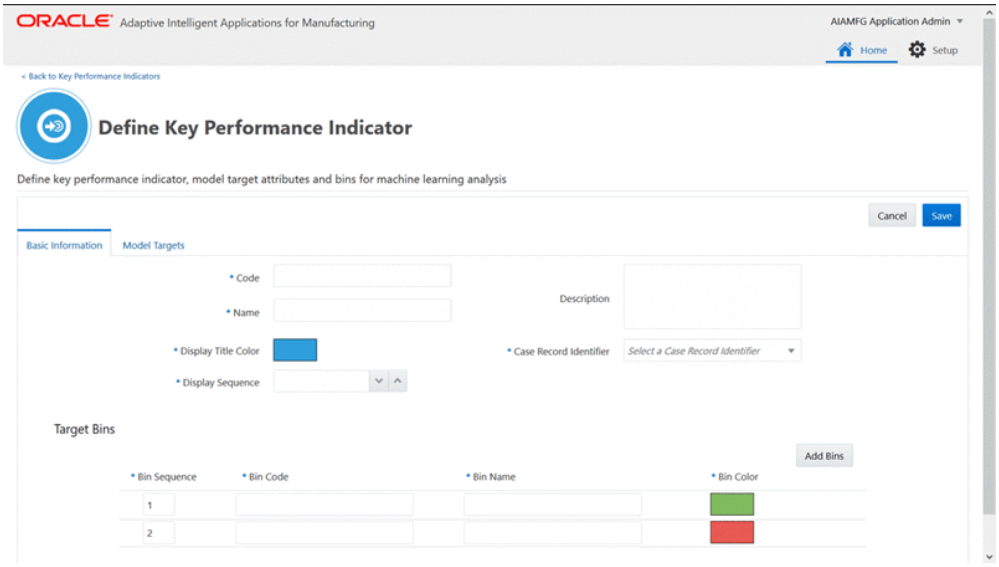
1. Navigate to the Key Performance Indicators page. From the Home Page, click **Insights** or **Predictions**. Click the **Configuration** link, then **Key Performance Indicators**.




2. You can view the existing Key Performance Indicators as tiles, each representing a KPI that appear in the Key Performance Indicator page. All the KPIs defined, both seeded and custom, display for both process manufacturing and discrete manufacturing organizations.

   KPI tiles appear in the order of the display sequence number you choose to set. If the display sequence number is common to one or more KPIs, then the tiles are arranged in ascending order of the KPI name.

3. To define a new KPI, click the **Plus** icon.




4. In the Define Key Performance Indicator page, use the Basic Information tab to add information for the following fields:

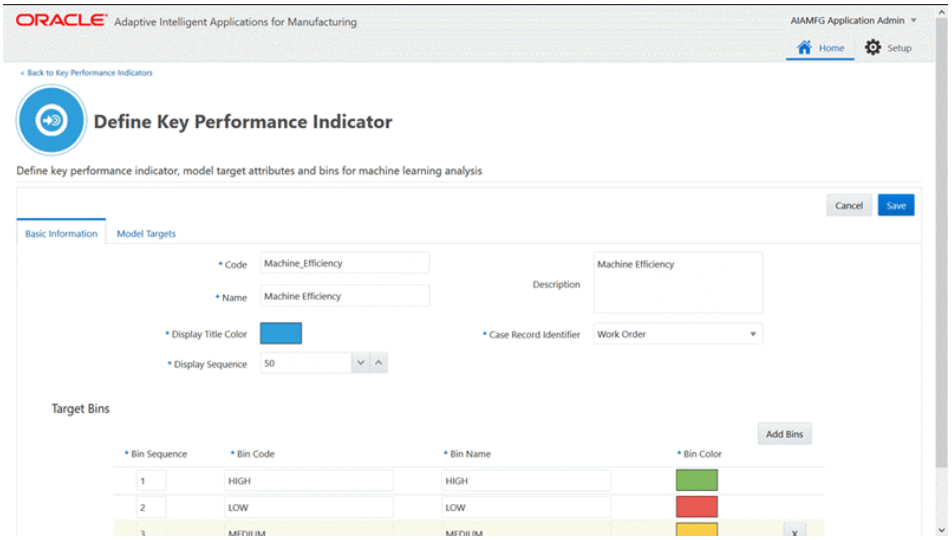
   - **Code** - Enter the KPI display code.
   - **Name** - Enter the KPI name. Note that you can only enter a name for a custom KPI and edit it till the output attribute is used by a dataset. Once it is used in dataset creation, this field will be read only.
   - **Description** - Enter a KPI description.
   - **Display Tile Color** - Select from the available color options to associate the KPI tile to a color.
   - **Display Sequence** - Select a value to arrange the KPI tile in a specific sequence in the ascending order.
   - **Case Record Identifier** - Select to specify if the KPI is for Work Order or Serial Unit level analysis.

5. Use the Target Bins region to add information for the following fields:

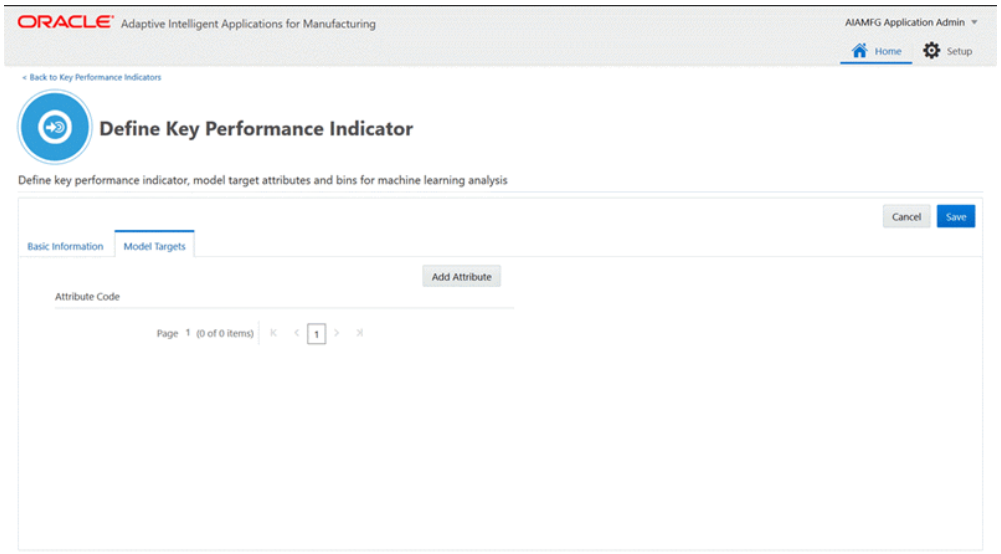
   - **Bin Sequence** - Enter a number from 1 to 5 and ensure it is unique within the KPI definition.
   - **Bin Code** - Ensure you enter a unique bin code for the KPI.
   - **Bin Name** - Ensure you enter a unique bin name for the KPI.
   - **Bin Color** - Ensure the color you select from the available options is unique

within the KPI definition.

You must define a minimum of two bins. To enter more than two bins, click **Add Bins** and enter information for the bin. You can add a maximum of five bins. You can change and delete bins until they have been used with a data set.



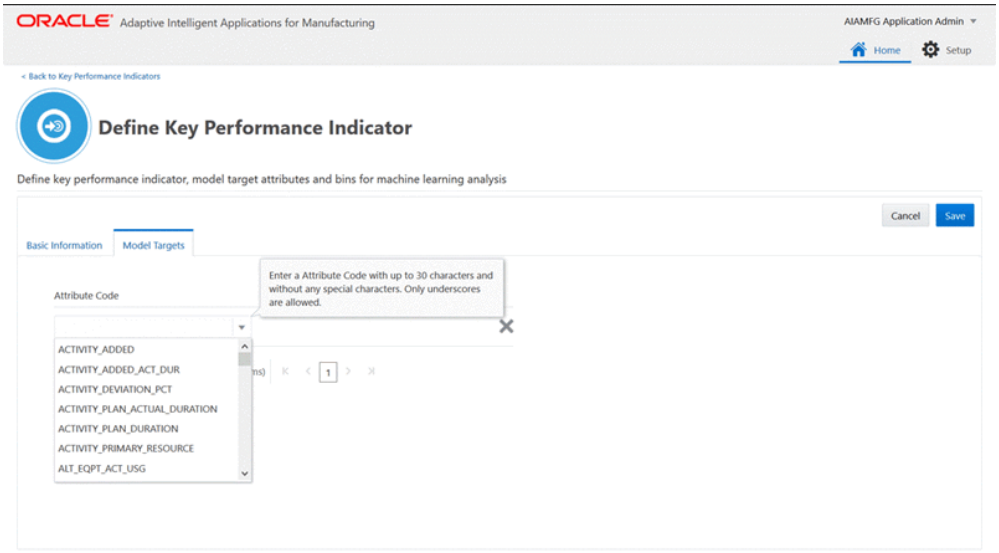

6. Optionally, to map a target to the KPI, in the Model Targets tab, click **Add Attribute** .




You can select a seeded attribute to associate and map to a KPI in the Model Targets tab or enter a custom attribute code. If you map an attribute to a KPI, then during dataset creation, when you select an attribute as a target in the Create Dataset user interface, the key performance indicator association will default from this mapping.

**Note:** Seeded attributes will not be available to be associated as targets for the Serial Unit Yield KPI.

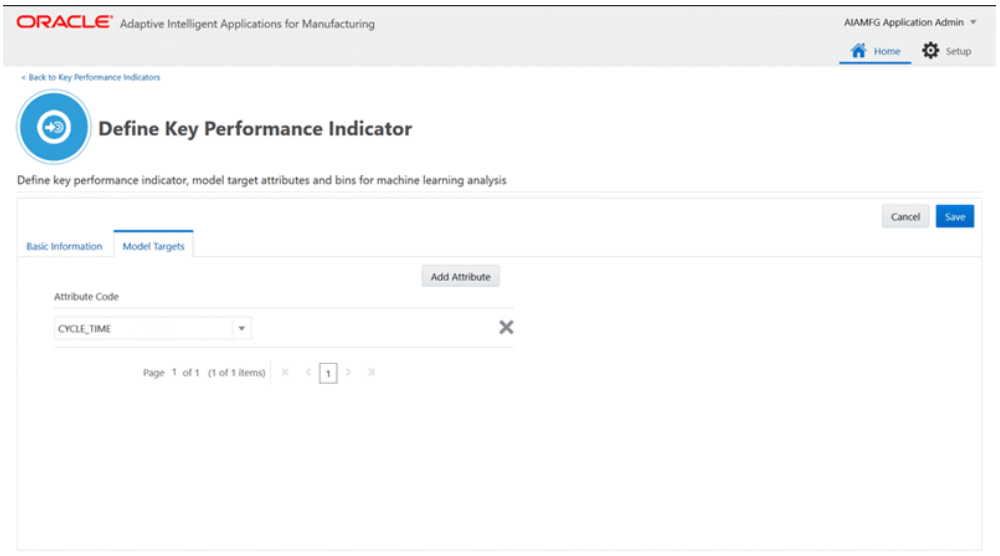
7. Select a seeded attribute code or enter a custom attribute code in the Attribute Code field.



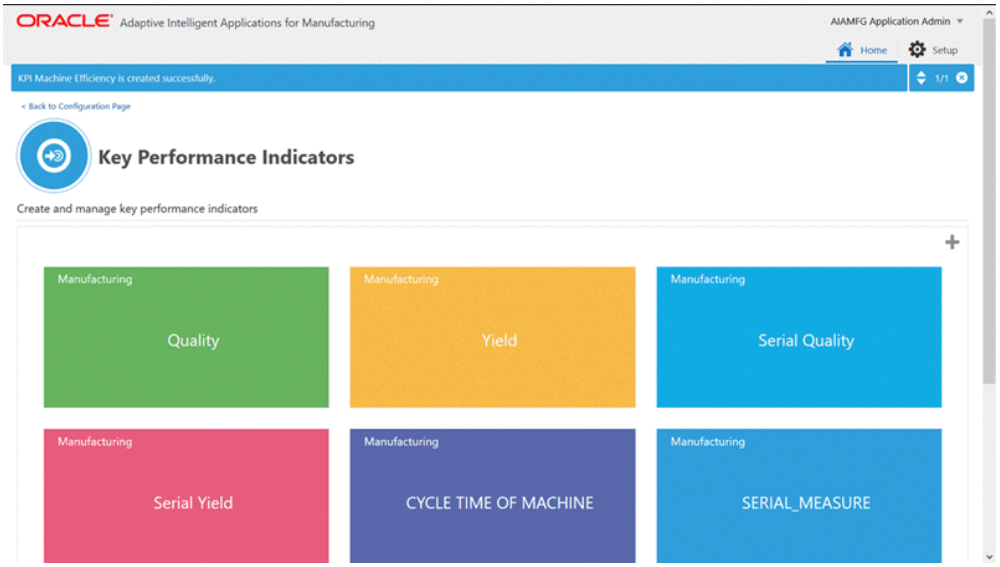

Note that once an attribute is selected, mapped to a KPI, and used in a dataset, the attribute cannot be deleted or associated with any other KPI.

If the attribute is not mapped during KPI definition, but is chosen as a target when creating a dataset and mapped to a KPI, then the attribute is automatically mapped to the KPI and is displayed in the Define Key Performance Indicator page for the KPI.

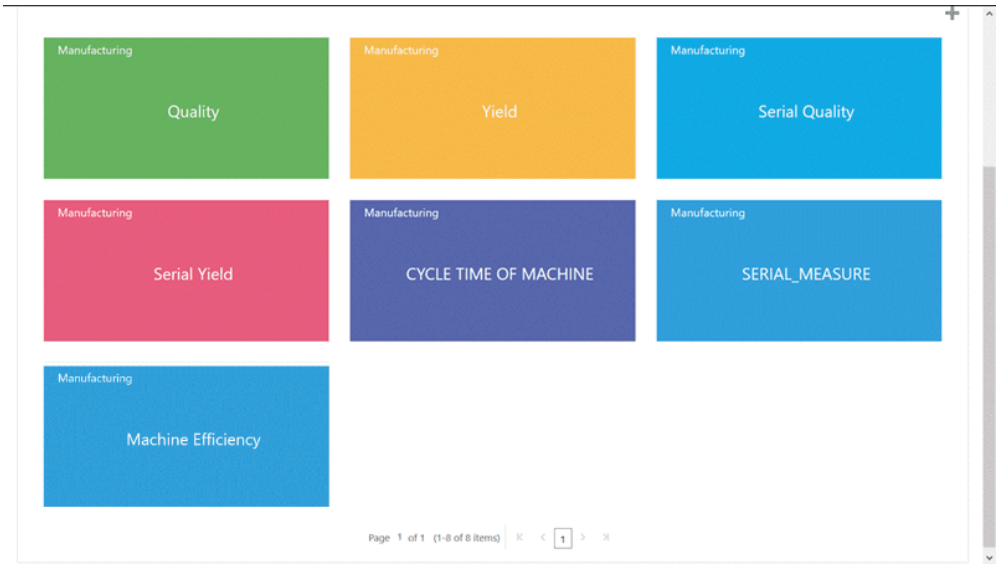
8. Click **Save**.




9. Once you save the details of the custom KPI, you will be returned to the Key Performance Indicators page where a message displays that the KPI has been successfully created.

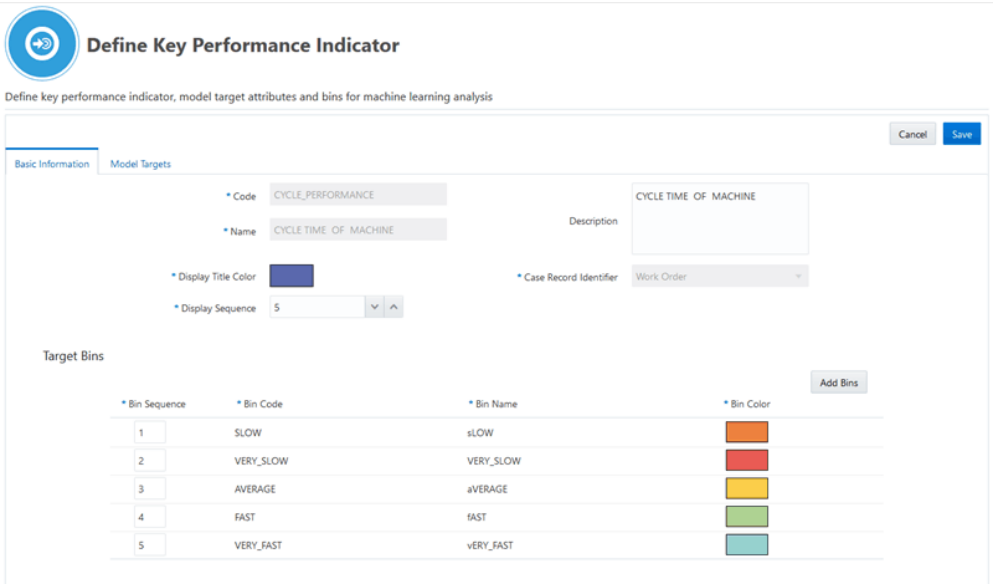



View the custom KPI you created which now appears as a tile in your selected display tile color. The tile position is based on the display sequence entered in the KPI definition.




### To edit a key performance indicator:

1. To edit a key performance indicator, in the Key Performance Indicators page, click the KPI tile you want to edit. The details of the KPI displays in the Define Key Performance Indicator page.




2. For custom KPIs and the seeded Yield, Quality, and Serial Unit Quality KPIs, you can edit the KPI depending on whether any of the targets associated to the KPI is used in a dataset.

   As long as any targets associated to a KPI are not used in a dataset, except for the

seeded Serial Unit Yield KPI, you can:

- Update KPI name, description, sequence, color.
- Add bins (only 5 bins are allowed).
- Remove bins.
- Update bin name, sequence, and color.
- Add or remove attributes.

If any targets associated to a KPI are used in a dataset, except for the seeded Serial Unit Yield KPI, you can:

- Update the KPI description, sequence, and color.
- Add bins (only 5 bins are allowed).
- Change bin sequence and color.
- Add attributes.

For the seeded Serial Unit Yield KPI, you cannot update the KPI definition and bin definition. You can add or delete a model target associated to the KPI, but you cannot delete a model target after the dataset is created. Any model target associated with the Serial Unit Yield KPI must have a value of 1, 2, 3, or 4.

3. Once you complete your updates for a KPI, click **Save**.

## Creating Datasets for Analysis

Specify context information when creating a dataset. The context information includes criteria to identify a subset of historical transactional data, such as the product, recipe, routing, and work order completion date range. This dataset submission establishes the data features and extracts the actual data from source systems.

The dataset you create performs two actions:

1. Extracts the out-of-the-box input features and targets such as operation duration, material quantities, quality results, resource usage, and custom features, defined as flex attributes. The input features and target attribute metadata information are extracted from all of the related ERP structural entities and time series data in the context of a product, recipe, routing and work order completion date range.
2. Extracts the data for the selected input features and target attributes.

You must create a dataset before creating a model. When you create a model, you specify which dataset to use as input for analysis.

Use the Data Preparation page to:

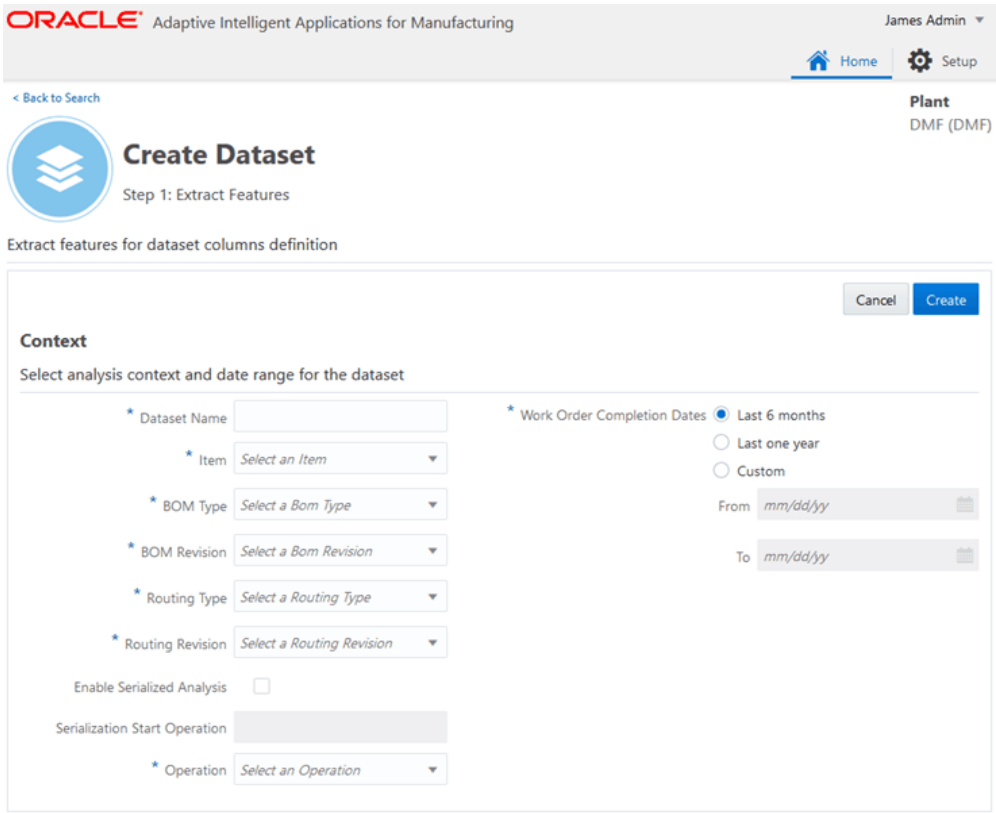
- Create new datasets.
- View existing dataset information.

### To create a dataset:

1. Navigate to the Data Preparation page.

   From the Home page, click **Insights** or **Predictions**, and then click the **Data Preparation** link.

2. In the Data Preparation page, click **Create** to create a dataset for analysis.




Creating a dataset involves the following two steps:

- Step 1: Extract Features. Select analysis context and date range for the dataset.

  This step defines the analysis context for a dataset and specifies the date range.

- Step 2: Select Target Attributes and Features. Select target attributes and features for dataset definition.

  Select the attributes that become target output measures and input features for

the dataset.

**Important:** You can only complete step 2 if work orders or serial units exist for the selected context in the step 1.

3. Begin by entering the following mandatory information in the Context section:

- **Dataset Name** - Enter the name of the dataset.
- **Item** - Select an assembly/production item from the list.

**For Process Manufacturing**

- **Item Revision** - Select an existing item revision from the list.
- **Recipe** - Select an existing recipe for the item/item revision from the list.
- **Recipe Version** - Select an recipe version for the item recipe from the list.
- **Operation** - Select an operation defined in the routing for the recipe selected above. All of the out-of-the-box features and flex attributes pertaining to all operations in the routing are extracted. See: Model Features for Process Manufacturing, page B-1.

**For Discrete Manufacturing**

- **BOM Type** - Select a primary or alternate BOM for the item from the list.
- **BOM Revision** - Select a BOM revision for the BOM type from the list.
- **Routing Type** - Select an existing item routing.
- **Routing Revision** - Select a routing revision for the routing type selected above.
- **Operation** - Select an operation defined in the routing selected above. All of the out-of-the-box features and flex attributes, up to and including this operation, are extracted. See: Model Features for Discrete Manufacturing, page B-13.

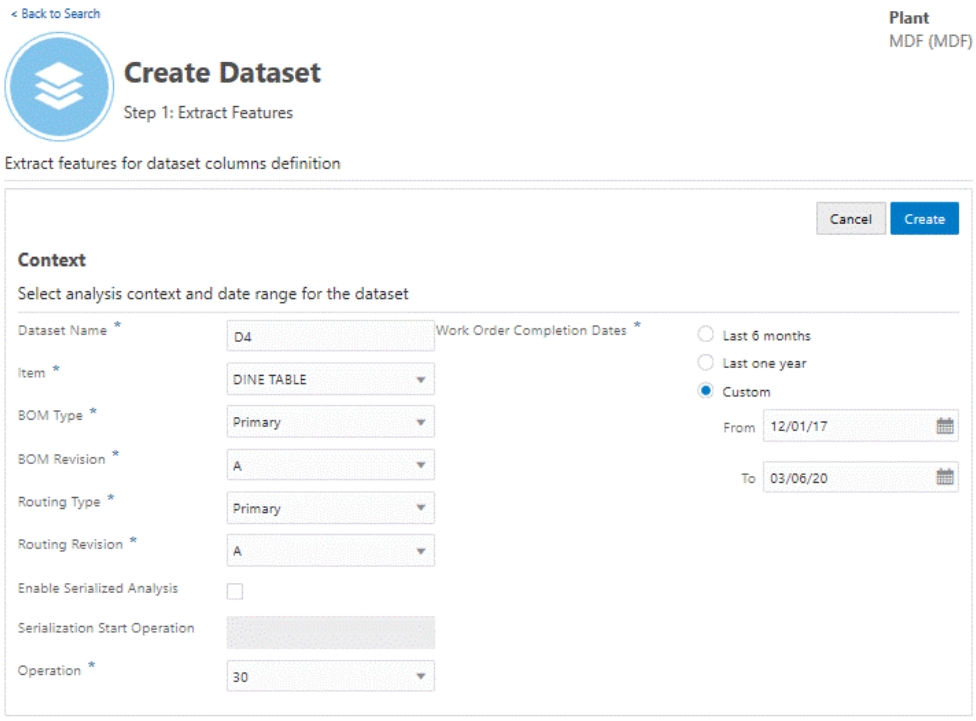
**Discrete Serialized Manufacturing-Only Fields**

- **Enable Serialized Analysis** - Automatically enabled if the item and routing type entered above are serialized. Optionally, you can disable serialized analysis if you want to predict results using an operation that occurs before the serialization start operation. If the serialization start operation is the first operation in the routing, then you can not disable serialized analysis.
- **Serialization Start Operation** - Automatically selected based on the

serialized item and routing type entered above. This is a display only field.

- **Work Order Completion Dates** - Select the date range of work orders by completion date that you want to analyze the data.

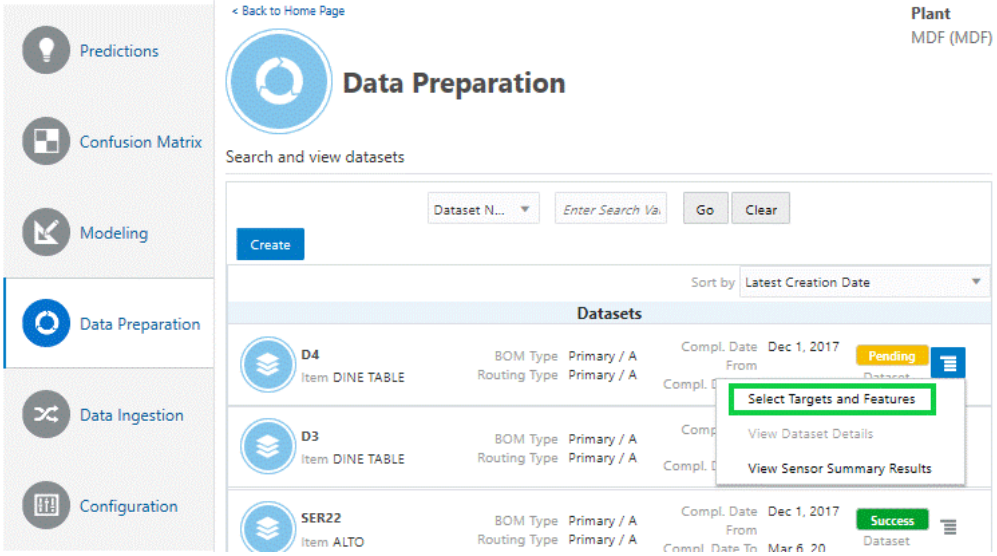
  **Additional Information:** If **Enabled Serialized Analysis** is checked, then this field name becomes **Serial Unit Completion Dates**.




4. Click **Cancel** to cancel the dataset creation request. Click **Create**.

   This submits a background request. The dataset is now listed in the Data Preparation page with a status for the Dataset as PENDING or the Feature Extraction as ERROR. If you receive an error status, then view the run details for the request from the Background Process page.

5. Select the Action link for your new dataset, then click **Select Targets and Features**.



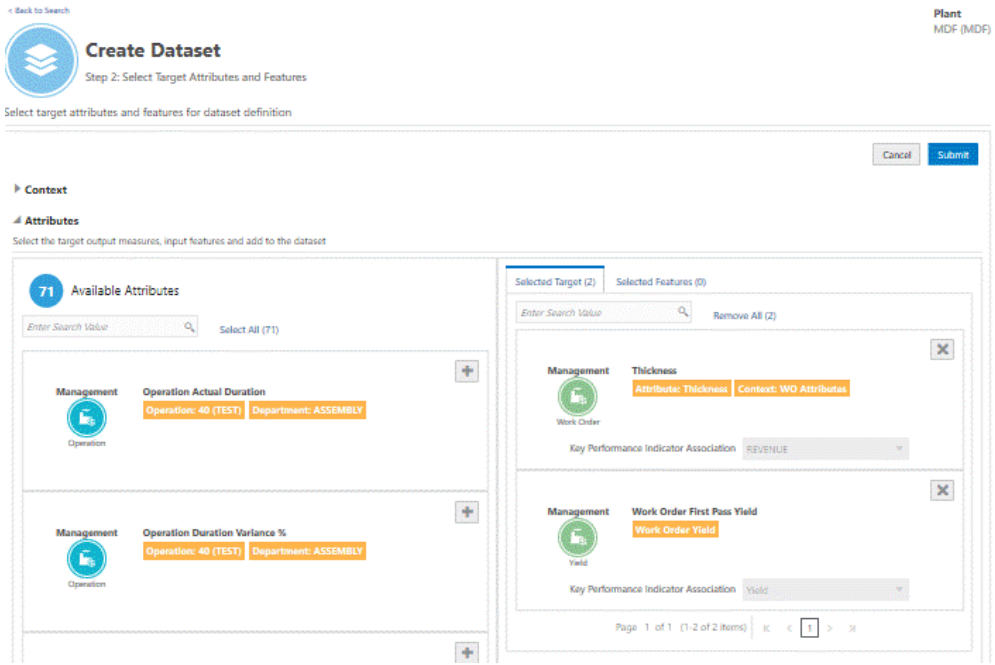

6. Select the attributes that you want to use as targets. On the right side of the Create Dataset page, verify that the Selected Targets tab is selected. On the left side, click the + icon next to each available attribute you want to use as a target output measure.

    - Search for attributes by:
        - Category
        - Subcategory
        - Key Performance Indicator
        - Attribute Name
        - Entity
    - You can only select an attribute as a target output measure if it is associated to a KPI and if the attribute has a numerical data type.
    - No more than 30 attributes can be selected as targets.
    - Only attributes with operations on or before the context operation are allowed as input features.
    - Only attributes with operations on or after the context operation are allowed as targets. Targets are by default

associated to a KPI according to the KPI definition. If a target is not associated to the KPI already, you can associate it to any of the seeded or custom KPI defined.

> **Note:** For the Serial Unit Yield KPI, you can associate only the seeded targets (Serial Unit Yield and Serial Unit Operation Yield) and custom attributes as long as they hold the distinct values, 1,2,3,4 in their data.

> **Tip:** If you use the Select All link, only the eligible attributes are selected as target output measures. You can use the X icon to remove an individual attribute as a target.




7. Select the Selected Features tab before adding attributes as input features. Click the + icon next to each available attribute you want to use as an input feature.

    - You can select up to 450 categorical features and 450 numerical features.

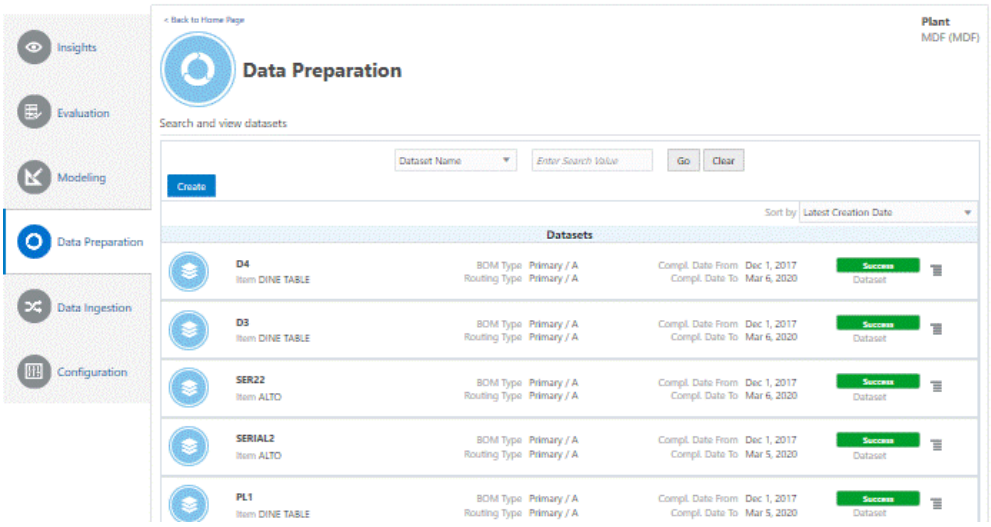
    - You can only use attributes with operations on or prior to the

context operation as input features.

8. Click **Cancel** to cancel step 2. Click **Submit**.

   After submitting the dataset background request, the dataset is displayed in the Data Preparation page with a dataset status as IN PROGRESS. When the dataset has been created, the status changes to SUCCESS. You can then use the dataset to create a model.

   If the dataset status changes to ERROR, navigate to the Background Process page to view the run details. See: Running Background Processes, page E-1.

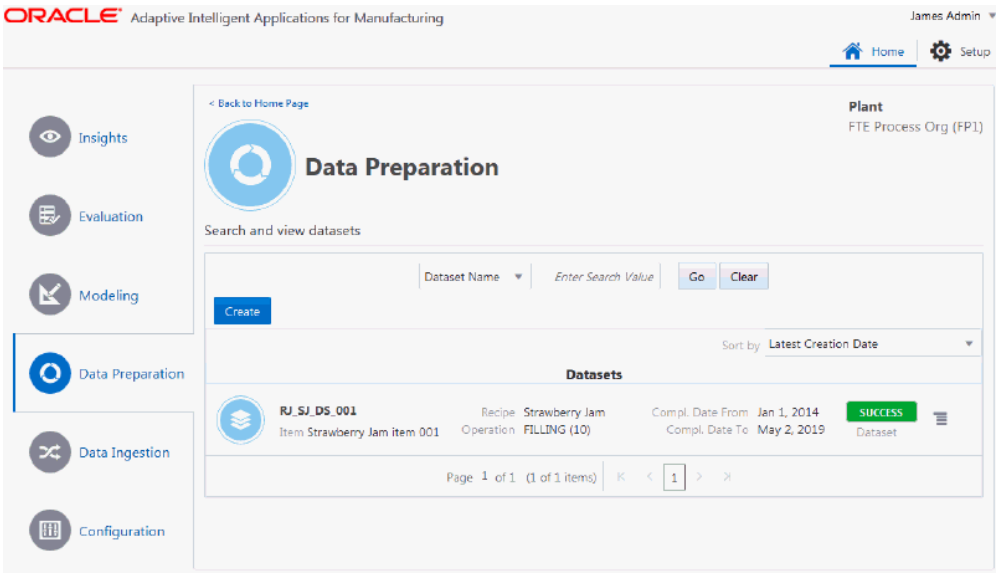



## To view dataset information:

1. Navigate to the Data Preparation page.

   From the Home page, click **Insights** or **Predictions**, and then click the **Data Preparation** link.

   The existing datasets display in the search results table of the Data Preparation page.




Use the Sort by field to sort the datasets by name or latest creation date.

You can also search for datasets using the following criteria, depending on the organization type:

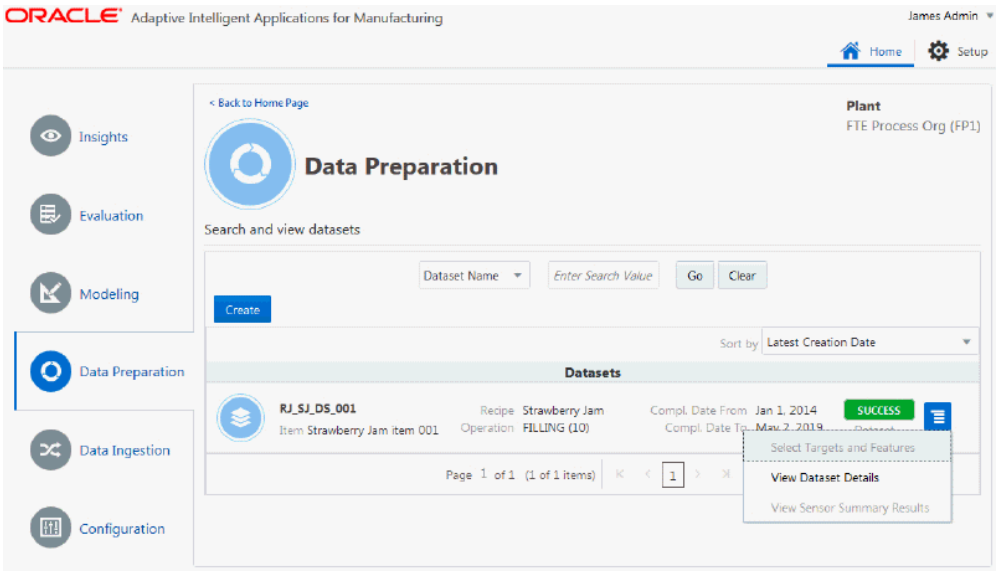
**Discrete Manufacturing Organization Criteria**

- Dataset Name
- Item
- BOM Type
- Routing Type
- Feature Extraction Status
- Dataset Status

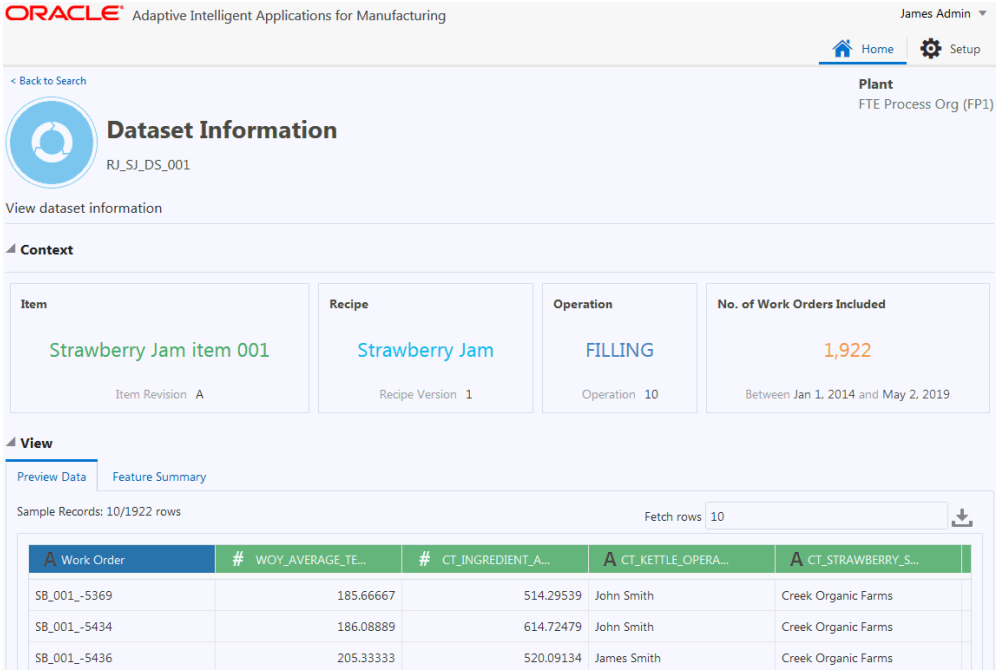
**Process Manufacturing Organization Criteria**

- Dataset Name
- Item
- Recipe
- Feature Extraction Status
- Dataset Status

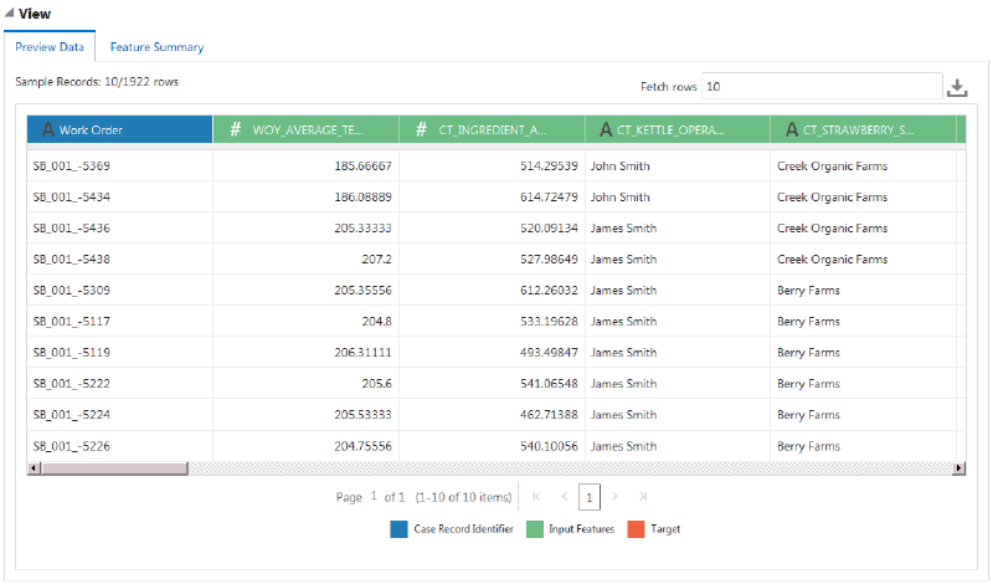
2. To view detailed information for a datatset, use the Action link for a specific dataset and click **View Dataset Details**.



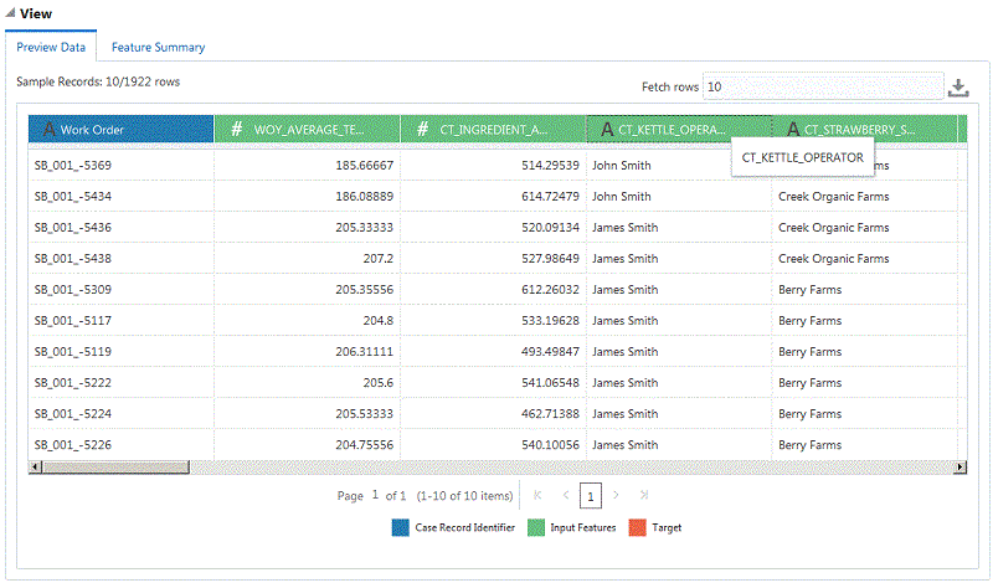

3. You can use the Dataset Information page to review details of the context information of the dataset you selected.




4. You can use the View region of the Dataset Information page to see the dataset details. The Preview Data tab displays case record identifiers, input features and targets for the dataset.

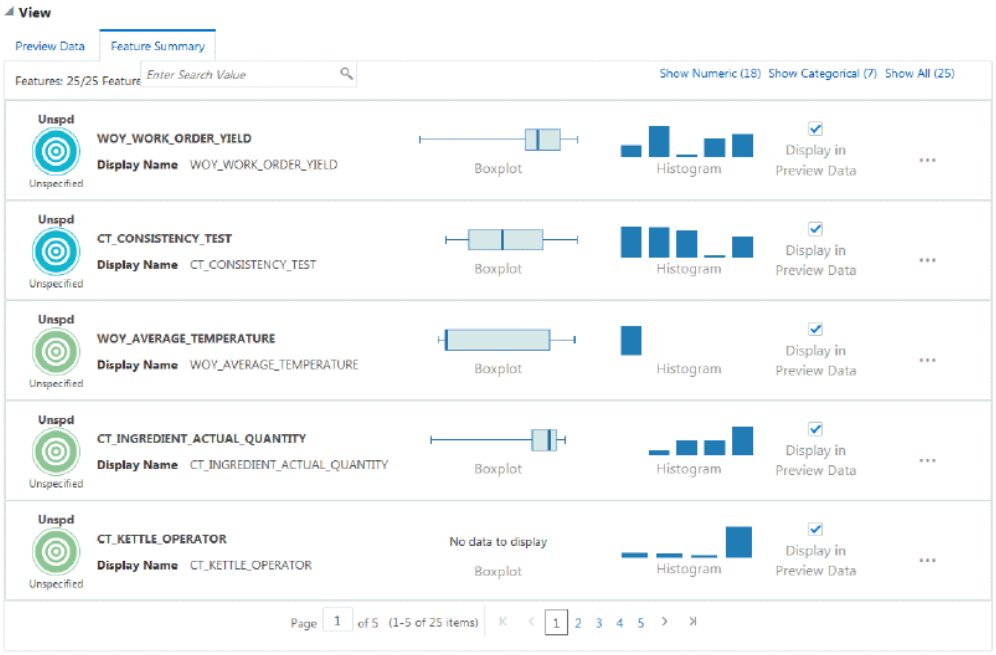



You can use the link in each column header to view additional details of the input feature or target attribute.




5. Use the Feature Summary tab to view the data distribution and details of input features and target attributes.

   Click in the Features field, then select a category, subcategory or feature from the drop-down list to narrow the list of features displayed.

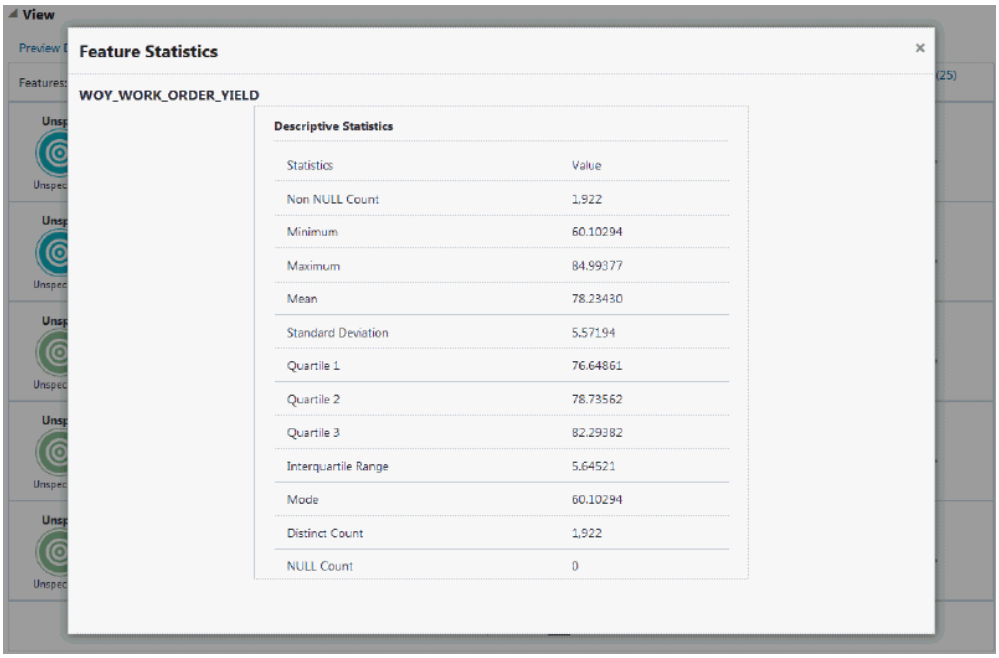



Click the Boxplot to learn which two quartiles contain the most data points.

Click the Histogram to discover the frequency distribution of data points across up to 10 frequency ranges.

You can select the input feature or target attribute you would like to appear in the Preview Data tab using the Display in Preview Data check box.

Click the ellipsis points (...) to view additional statistics for an input feature or target attribute.



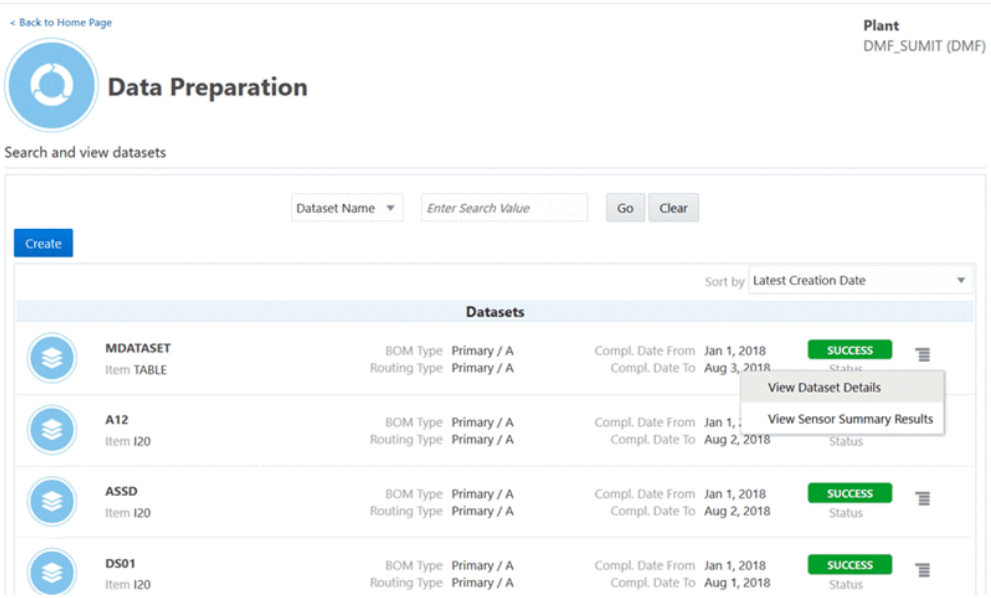

## Viewing Sensor Summary Results

After the successful creation of dataset, you can view the time series features generated according to the time segments and functions applied to the contextualized sensor stream data used in the request. The time series features are created for all equipment process parameters with an assigned time series feature set. You can drill into a specific process parameter to understand and compare the summary function values of features across the work orders in the context.

### To view sensor time series features:

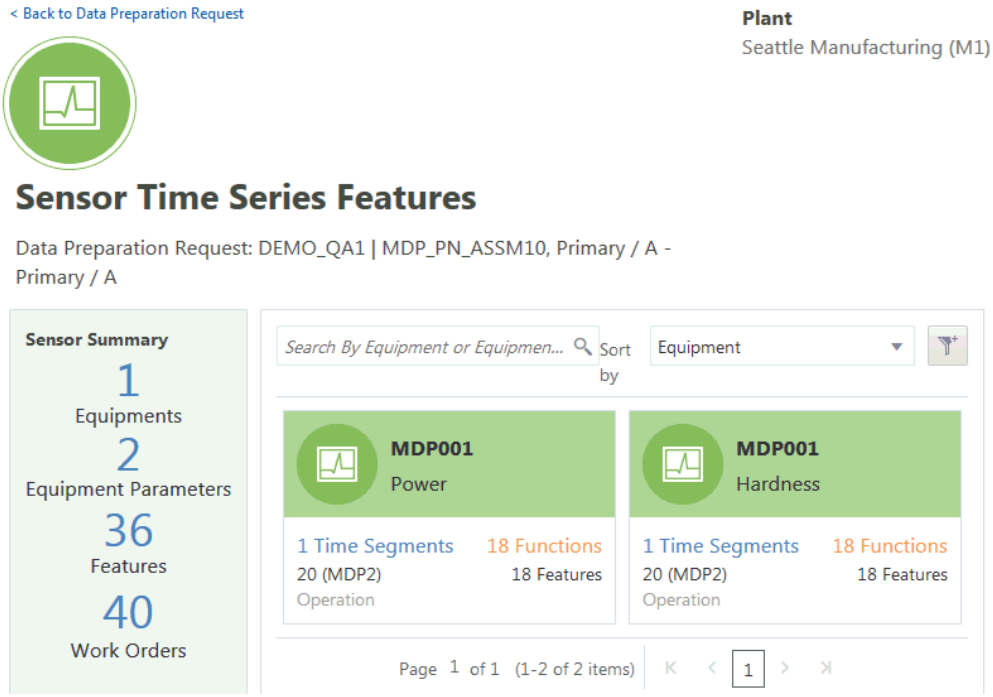
1. From the Home Page, click **Insights** or **Predictions**, and then click the **Data Preparation** link.

2. In the Data Preparation page, the existing datasets appear in the search results table. To view time series features for a dataset, click **View Sensor Summary Results** from the Actions link.




3. In the Sensor Time Series Features page, you can sort the results by Equipment, Equipment Parameter, Time Segments, and Functions.

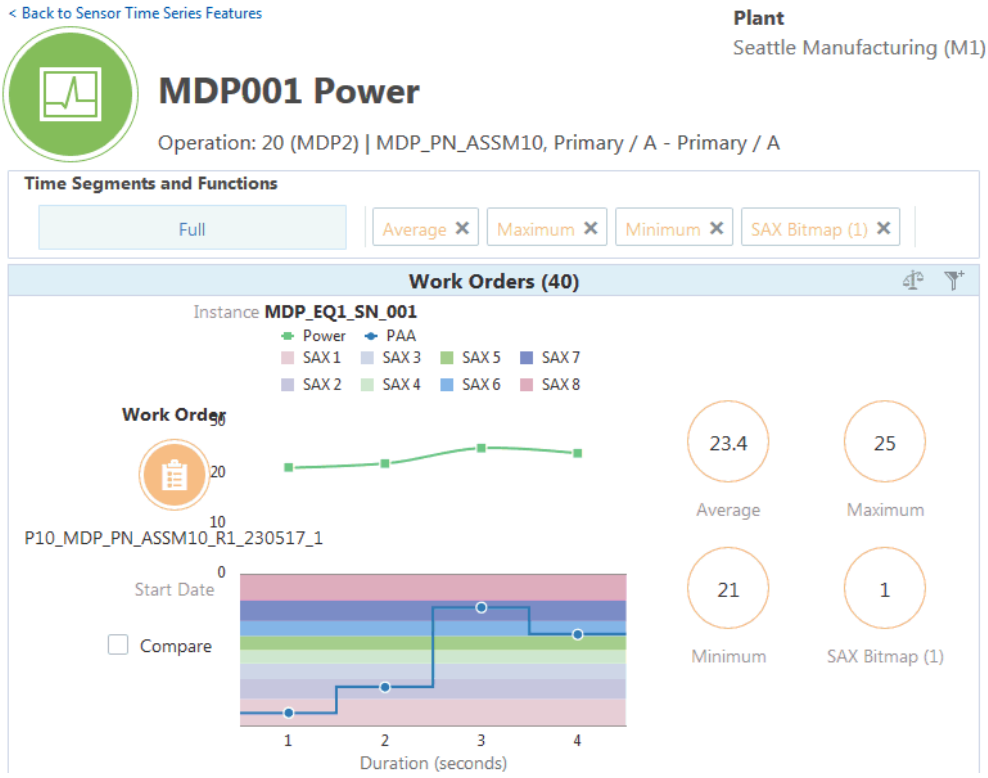
4. You can view the number of time segments, the number of functions, and the total number of features for an equipment instance and equipment parameter based on the time series features set definition.




### To view sensory summary results:

Click on any parameter to review the sensor summary results. This page enables you to filter by a time segment and functions defined for that parameter. This enables you to review the computed sensor summary values across work orders or serial units considered in the context.

1. View sensor time series data points for a selected time segment.
2. View the Symbolic Aggregate approximation (SAX) format of the sensor time series data for each work order or serial unit in the context. This helps you to visually understand the patterns in the time series data.
3. View the sensor summary function values for each work order or serial unit in the context.
4. Select multiple work orders and click on the compare icon to view and compare the sensor summary function values across the selected work orders or serial units.


< Back to Sensor Time Series Features
Plant
Seattle Manufacturing (M1)
MDP001 Power
Operation: 20 (MDP2) | MDP_PN_ASSM10, Primary / A - Primary / A
Time Segments and Functions
Full
Average
Maximum
Minimum
SAX Bitmap (1)
Work Orders (40)
Instance MDP_EQ1_SN_001
Power
PAA
SAX 1
SAX 3
SAX 5
SAX 7
SAX 2
SAX 4
SAX 6
SAX 8
Work Order
30
20
10
0
P10_MDP_PN_ASSM10_R1_230517_1
Start Date
Compare
1
2
3
4
Duration (seconds)
23.4
Average
25
Maximum
21
Minimum
1
SAX Bitmap (1)

# 6

# Analyzing Data

This chapter covers the following topics:

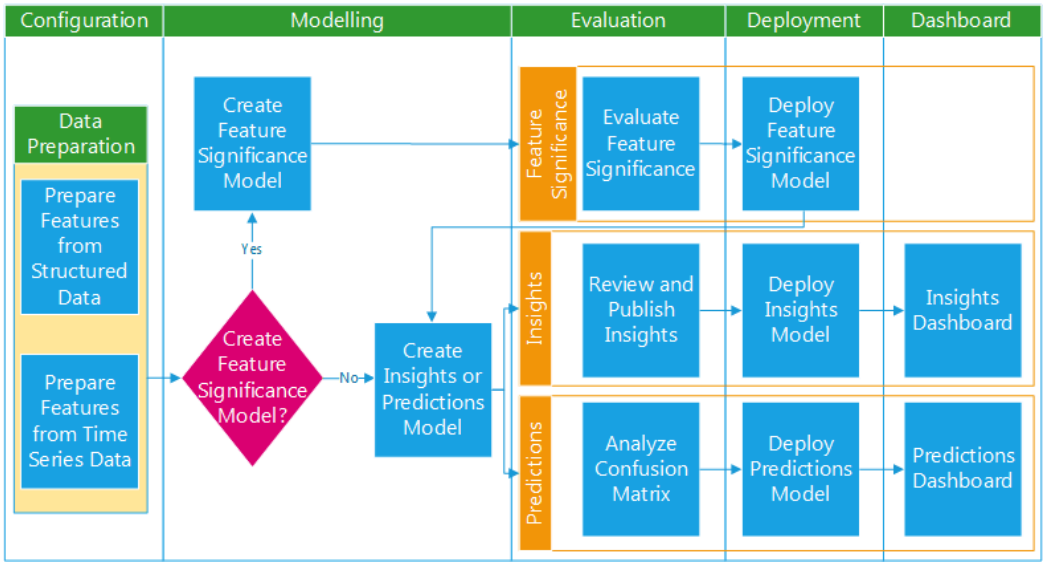
- Overview of Analyzing Data
- Creating a Model
- Evaluating a Model
- Deploying a Model

## Overview of Analyzing Data

The process of successfully creating a dataset extracts features from structured data and time series data. You can then analyze the data by applying data mining techniques to form deeper insights into improving target measures for products. The data mining process typically starts with building a statistical model that uses an algorithm on a set of data which contains a set of input features and a specific target variable. The model building process consists of the following steps:

1. Creating a model.
2. Evaluating a model.
3. Deploying a model.

***Model Building Flow***




## Analyzing Insights or Predictions Models Using Custom Key Performance Indicators

When creating an insights or predictions model, you have the option to select target measures associated with custom key performance indicators (KPIs). This enables you to analyze custom KPIs such as Cycle Time, Machine Efficiency, Machine Downtime and so on.

These target measures defined as custom KPIs can also be used in models to analyze the effect of flex attributes associated with business entities, such as the lot expiry date associated with a lot or the voltage associated with an operation.

## Related Topics

Defining Key Performance Indicators, page 5-22

Overview of Importing Business Entity Data, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*

# Creating a Model

You can build different models to achieve different purposes, such as understanding the significant features or analyzing the relationship between a specific target measure and the features selected. When you create a model you select the type of model, the target measure, features, algorithm, and deployment options. The model then creates a data set by extracting data from different work orders for the selected features and the output target measure, running the algorithm on the data extracted.

You can choose from the following three analysis types:

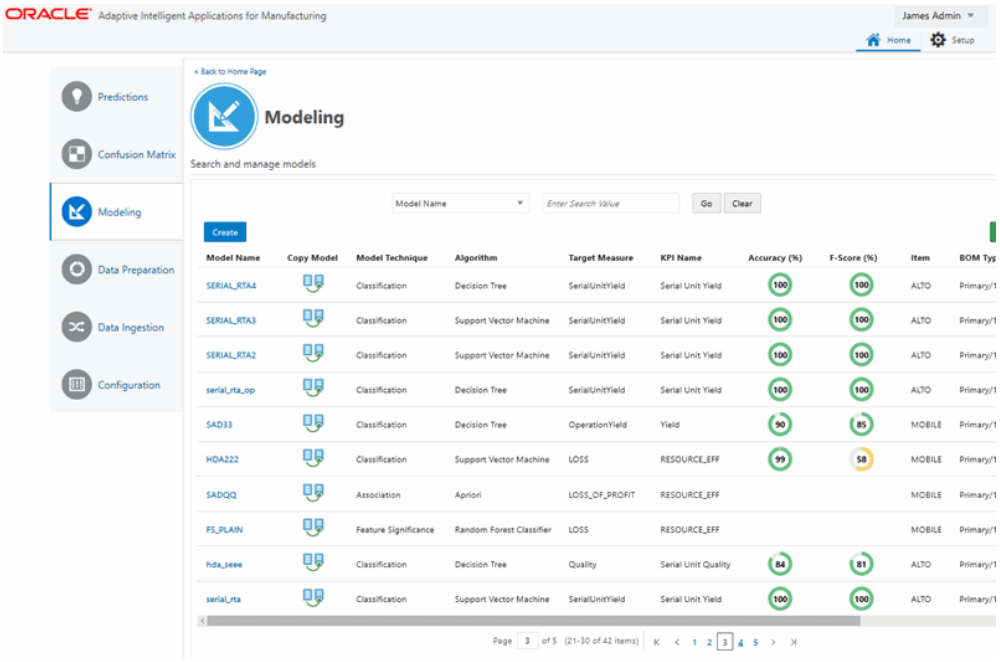
- Feature significance model - analyzes data and extracts the most influential features that potentially impact a given target measure.
- Insights model - identify patterns and correlations between the influencing factors and the output target measure. You can either select features from the ranked features generated from the Feature Significance model or select features that you are interested in for the analysis.
- Predictions model - identifies conditions under which predictions are made for a target variable by considering various input variables/predictors. You can either select features from the ranked features generated out of the Feature Significance model or select features that you are interested in for the analysis.

### To create a model:

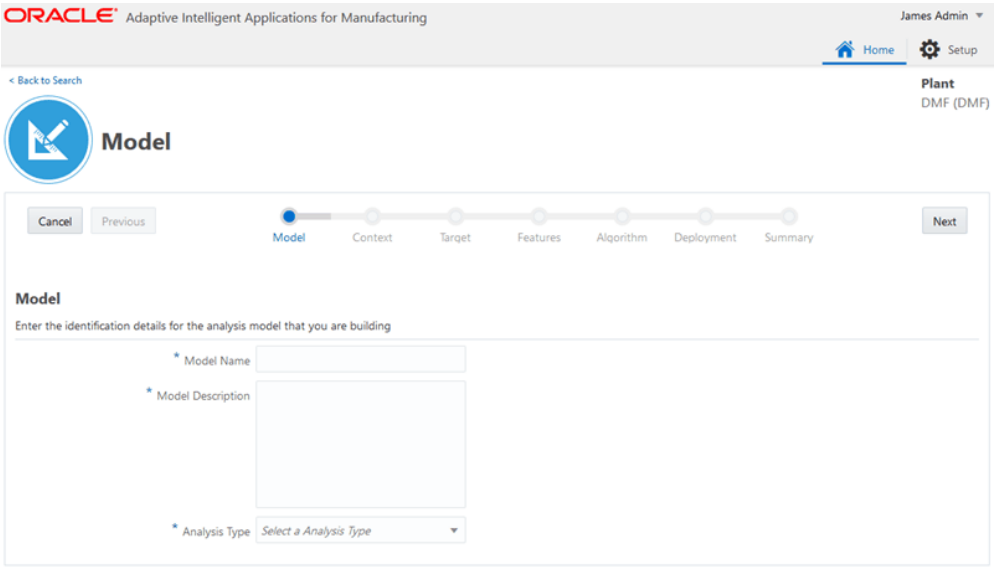
1. Navigate to the Model page.

   From the Home page, click **Predictions** or **Insights**, and then **Modeling**.

   The existing models appear in the Model page.

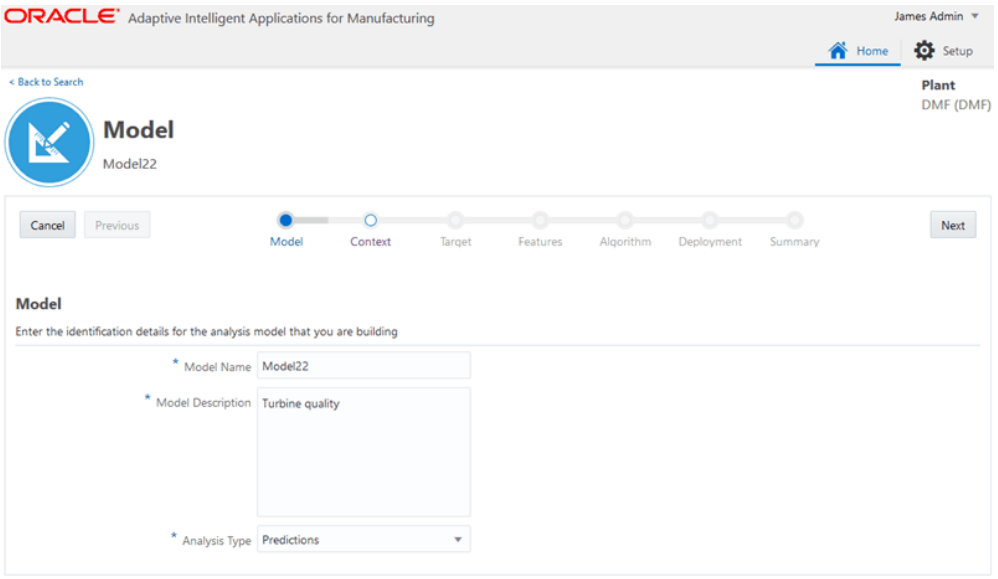



2. Click **Create**.

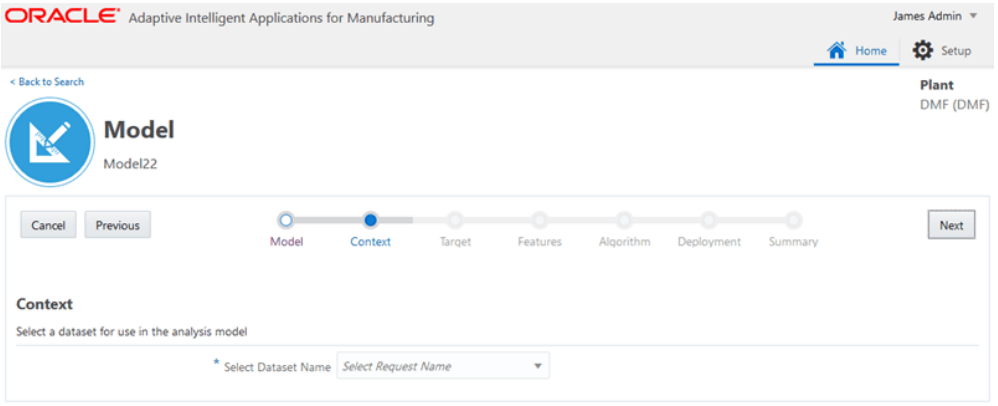



3. Enter the following model details:

   - **Model Name** - use only alphanumeric characters and the special characters #, $, and _.
   - **Model Description**
   - **Analysis Type** - select the type of model to create.
     - Feature Significance
     - Insights
     - Predictions

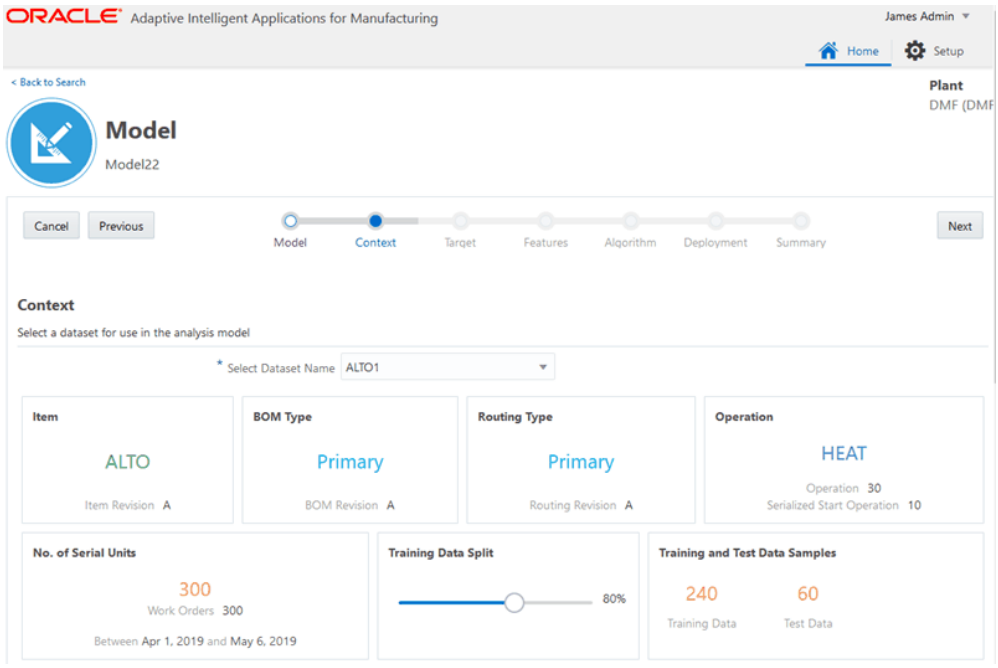



4. Click **Cancel** to cancel the process. To continue, click **Next**.



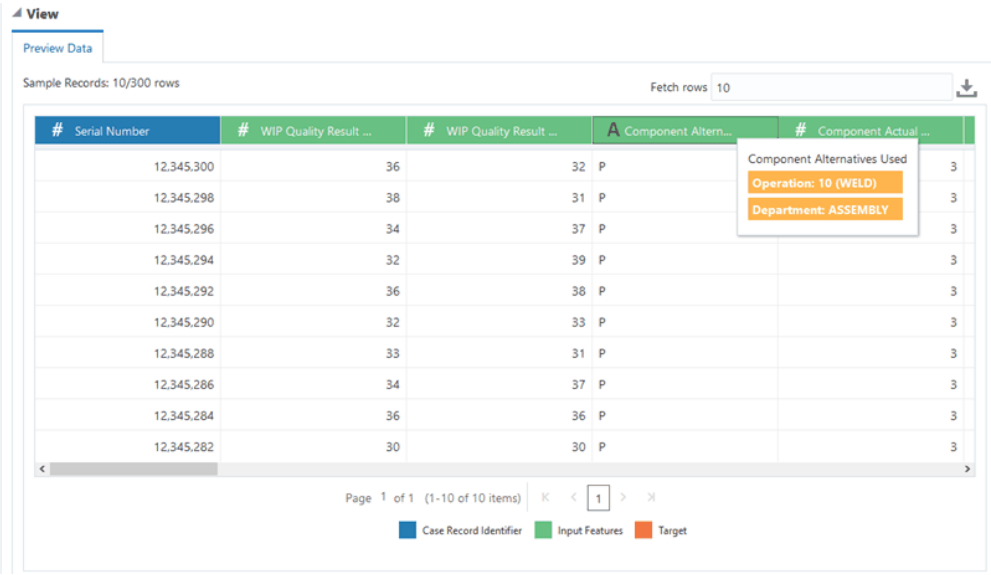

5. **Select Dataset Name** from the list in the Context section.

   Choose the request name of a dataset that you previously created.




After you select a dataset, you can scroll to view information in the Preview Data tab.

> **Tip:** Preview the data for multiple datasets. Use the **Select Dataset Name** field to select and preview various datasets while on this page.




**For Predictions models only**

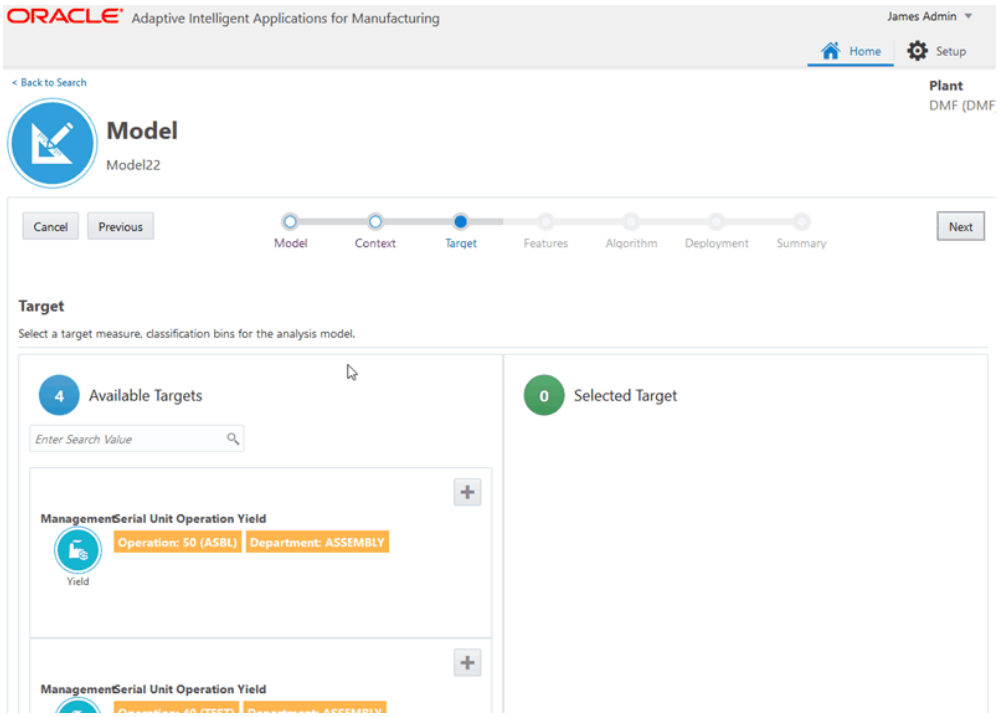
1. Specify the **Training Data Split**.

   Use the slider to split the training and test data from the sample. The Predictions module uses the training data to train and build the model and uses the test data to evaluate the model performance. The default slider setting is 80% training data and 20% test data. However, you can use the slider to select a training sample between 50-100% of the input data.

   The Predictions module uses stratified sampling to proportionately split data into training and test samples. This homogeneous splitting of the data ensures equal representation of the data in both training and test samples.

6. Click **Cancel** to cancel the process. Click **Previous** to go back to the Model step. To continue, click **Next**.

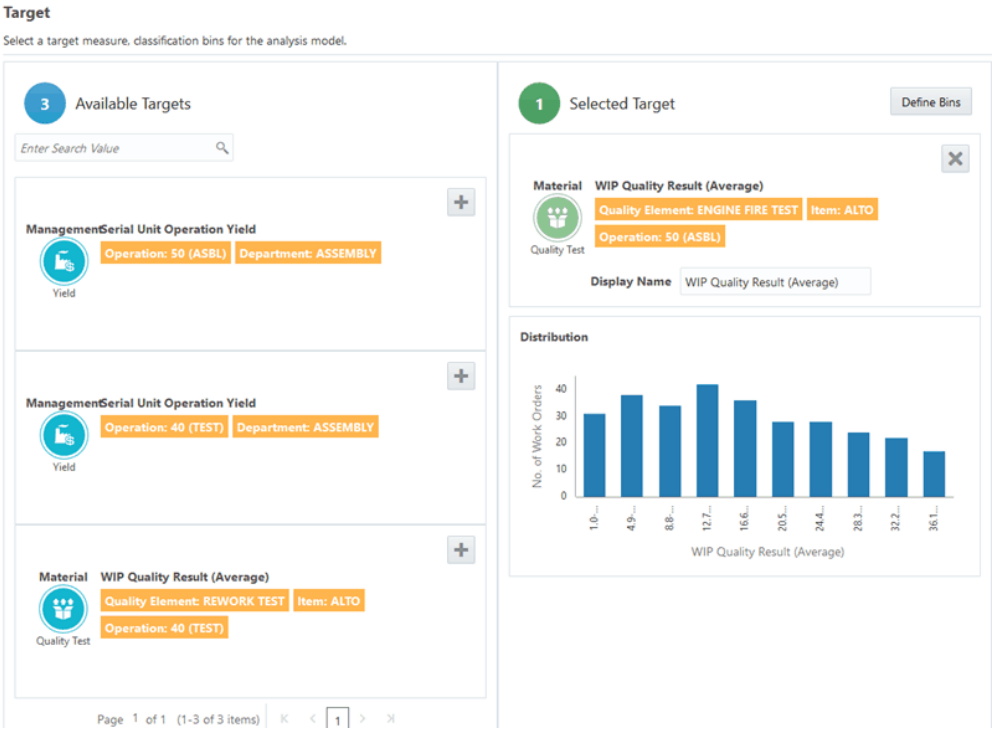
7. Select a target measure for the model from the list of available targets.




8. Optionally, change the selected target **Display Name**.

9. Define bin limits if none exist. Edit the bin limits, if needed. Click either **Define Bins** or the **Edit** icon.

    - Serialized analysis models with a yield target display the following seeded bins:
        - Feature Significance models: Pass and Fail bins.
        - Predictions models: Pass and Fail bins.
        - Insights models: Accepted With First Pass, Pass With Rework, Rejected, and Scrapped bins.
    - For a Predictions model, you have the option to turn on alerts on the Bins page for certain classification ranges. If actual results fall within these classification ranges, predictions/alerts appear on the Predictive Analysis page. See: Using Predictions Analysis, page 7-8.
    - Define a minimum of two bins for all other types of targets.




10. Add a new classification label and range, delete an existing classification label, or update the range of an existing label. Optionally, click **Clear Bins** to clear the bin range selection and remove the bin coloring from the charts.

    The following charts appear in the Update Bins page:

    - Bins: Displays the total number of case records.
    - Box Plot: Displays the distribution and quartiles.
    - Scatter Plot: Displays the distribution of target attribute data points and plots these points based on the case record day scale along the Y axis.

      **Note:** If there are more than 10,000 data points, stratified sampling is applied before displaying the scatter plot.

    - Histogram: Displays the frequency distribution of data points. Up to 10 frequency ranges are used to plot the histogram.

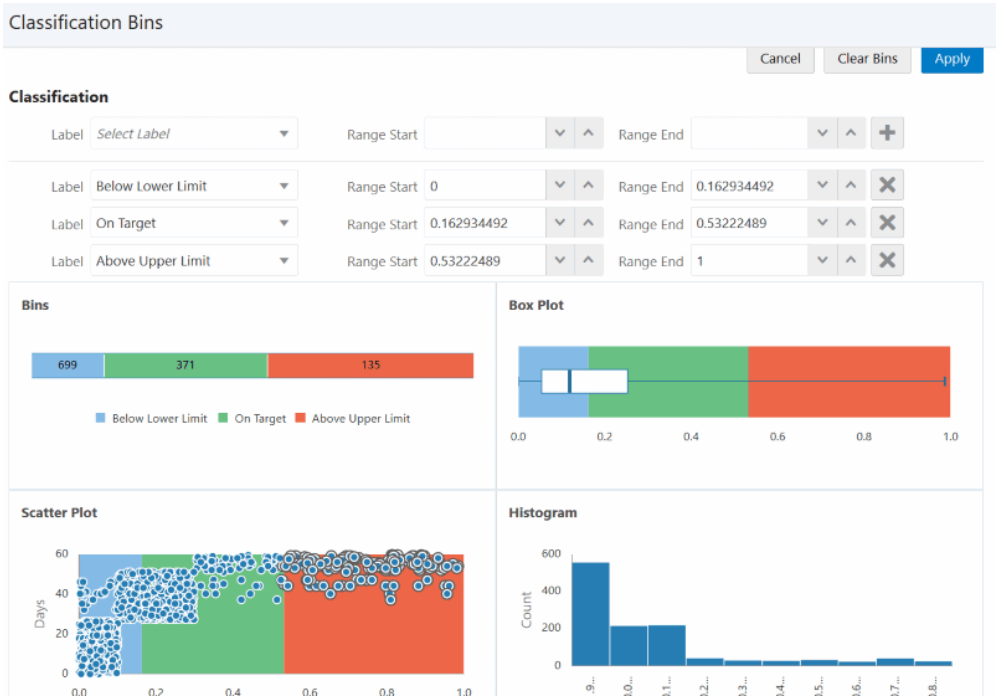
    Notice that the Bins, Box Plot, Scatter Plot, and Histogram charts change along with any classification changes made. Use the plus icon to add at least two classification ranges. The model predicts outcomes based on the ranges defined. For example, the model predicts a classification range, such as Low, Medium, or High, not a specific number, such as 84.6.

- Add classification ranges using the pre-defined labels provided.
- You can save the classification bins when:
  - No gaps exist between the ranges. An unclassified range (gap) appears as white space in the Bins, Box Plot, and Scatter Plot charts.
  - Each data point falls within a classification range.
  - Each classification range contains at least one data point.
  - Classification ranges do not overlap.

    **Tip:** To view the number of data points in each new or updated classification range, click the Show Count link in the Bins chart region.

- A data point is classified under a classification range if the Range Start <= data point < Range End, except for the last classification range where range end is also considered. For example, if you define the Low range as 0 to 20 , the Medium range as 20 to 80 and the High range as 80 to 100, then a result of 20 is classified in the Medium range and a result of 100 is classified in the High range.

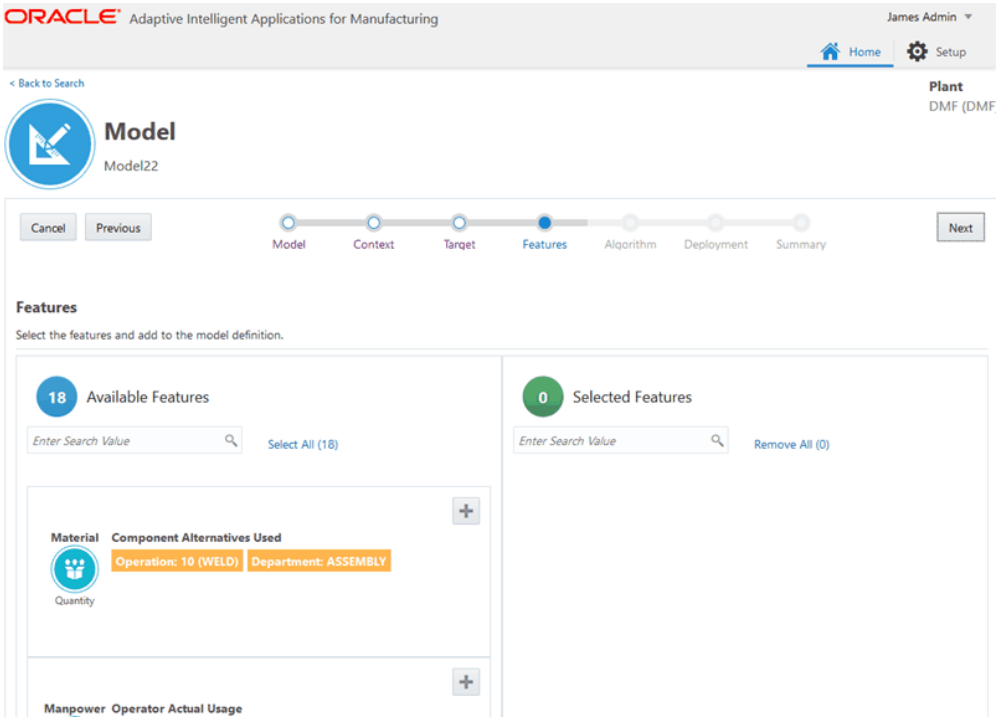
  **Tip:** Define a new classification range by dragging a selection area over unclassified data points in the Scatter Plot or use the Histogram as a visual aid while entering ranges.




11. Click **Apply**.

    The system returns to the Model page, Target step.

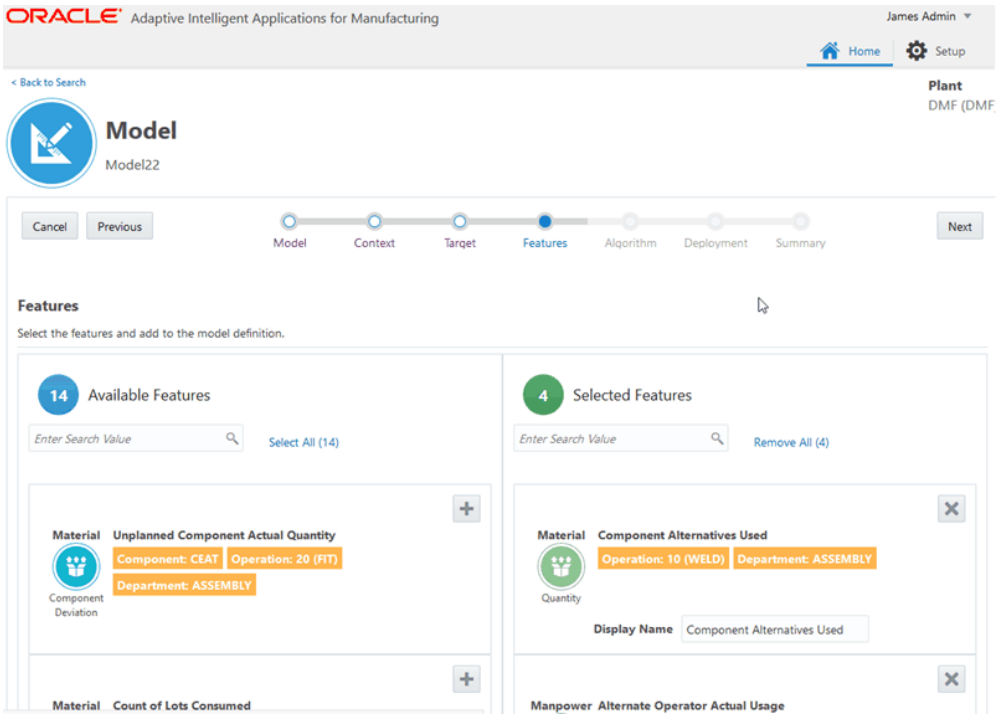
12. Click **Cancel** to cancel the process. Click **Previous** to go back to the Context step. To continue, click **Next**.




13. Search for and select the features to use as predictors in the model.

    Refer to Model Predictors for Process Manufacturing, page B-1 or Model Predictors for Discrete Manufacturing, page B-13 for a description of each feature.

    - Use the **Available Features** field to search for features.
    - Each feature may appear as an available feature multiple times, depending on the occurrence of the feature in the operations. For example, an operation duration feature appears for multiple operations.
    - Click the Feature Significance link for a feature to view the feature's statistics.
    - If you previously deployed a feature significance model that used the same context (dataset) as the Insights or Predictions model you are creating, then the **Ranked Features** box is checked so you can easily select the ranked features as predictors. Deselect **Ranked Features** to view all other available features extracted in the dataset.

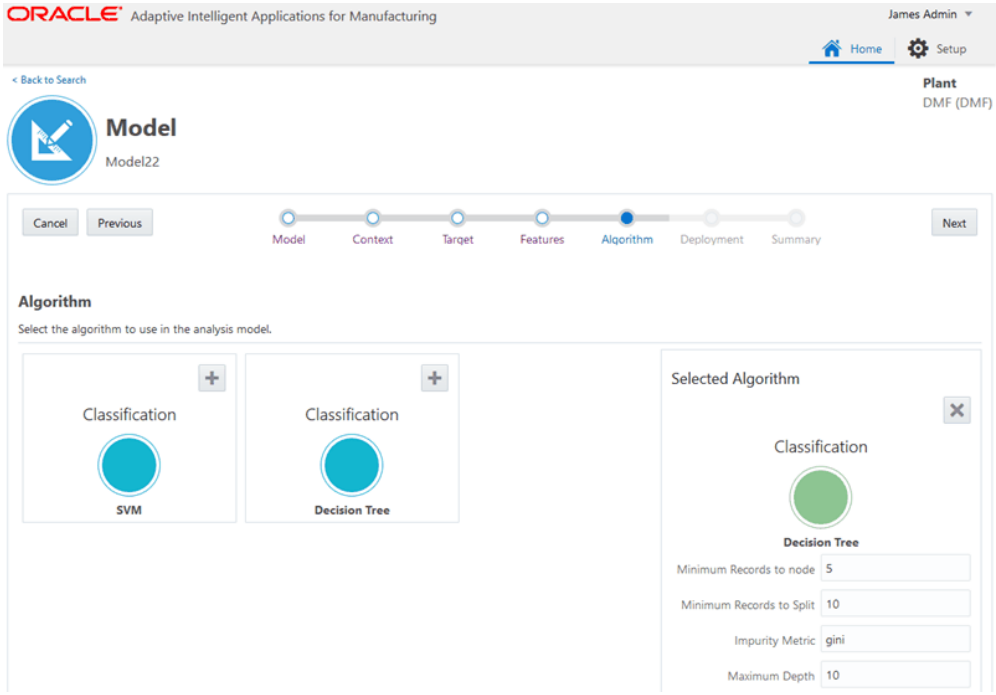



14. Click **Cancel** to cancel the process. Click **Previous** to go back to the Target step. To continue, click **Next**.

15. Select an algorithm for the model.

    **Note:** The algorithm choices vary depending on the model analysis type selected and certain algorithms have parameters that you can change. For more information about a particular algorithm, see:

    - Feature Significance Model Algorithms, page C-1
    - Insight Model Algorithms, page C-3
    - Predictions Model Algorithms, page C-4




16. Click **Cancel** to cancel the process. Click **Previous** to go back to the Features step. To continue, click **Next**.

17. Select whether or not to deploy the model.

    Selecting **Auto Deploy** deploys the model once it is built. You can only auto-deploy Feature Significance and Insights models. These two types of models only run once after deployment. If a Feature Significance or Insights model has been previously created using the same context and deployed, then **Auto Deploy** defaults to off.

    For a process manufacturing Predictions model, select from the following deployment options:

    - When to start the Prediction?
      - Step status becomes In Process - start when the step begins.
      - Step status becomes Pending - start when the batch begins.

        **Warning:** Do not select this deployment option when using the case record data ingestion method. This ingestion method only brings in historical work orders and in progress work orders, not pending work orders, or operations that are pending.

- x minutes after Step status becomes In Process - start x minutes after the step begins.

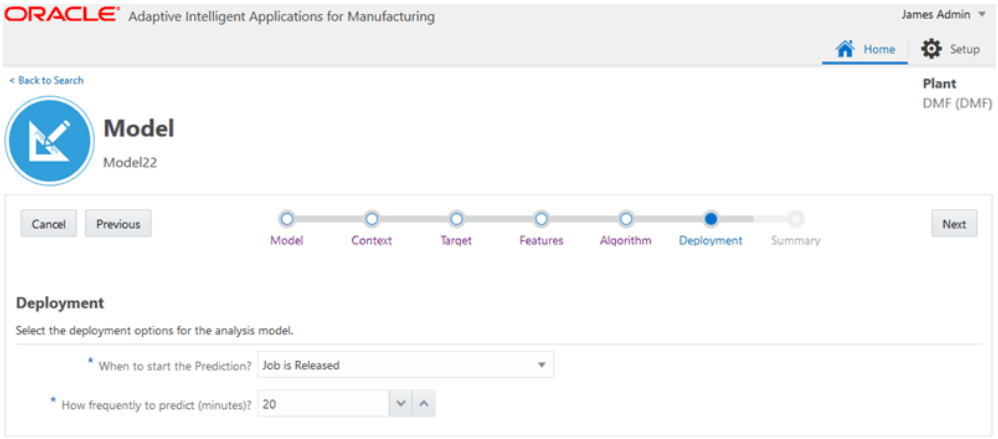
- How frequently to Predict? - select the frequency of prediction in minutes.

For a discrete manufacturing Predictions model, select from the following deployment options:

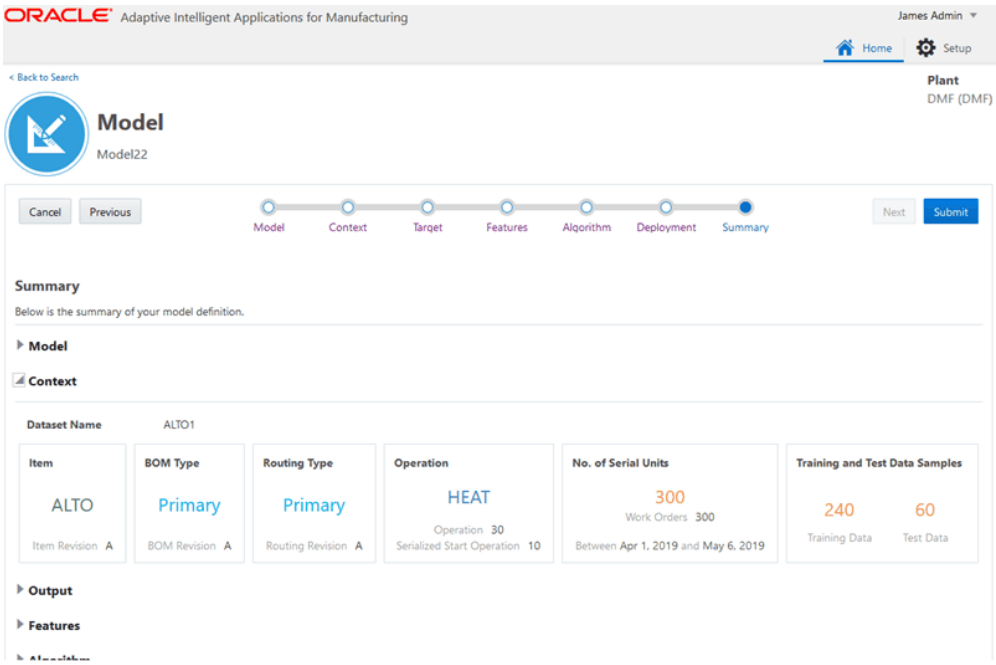
- When to start the Prediction?
  - Job is released - start when the job is released.
  - Assembly is in process at Operation - start when the operation begins.
  - Assembly is at Operation - start before the operation begins.

    **Warning:** Do not select this deployment option when using the case record data ingestion method. This ingestion method only brings in historical work orders and in progress work orders, not pending work orders, or operations that are pending.

- How frequently to Predict? - select the frequency of prediction in minutes.




18. Click **Cancel** to cancel the process. Click **Previous** to go back to the Algorithm step. To continue, click **Next**.




19. Review the model definition. You can choose to redefine the model using the previous pages.

    Click **Cancel** to cancel the process. Click **Previous** to go back to the Deployment step. To continue, click **Submit**.

20. When you submit the model, the system returns you to the Modeling page and the status of your request appears as IN PROGRESS.

    Confirm that the status of the model changes to COMPLETED. Should the model status be in ERROR, navigate to the Background Process page to view the run details. See: Running Background Processes, page E-1.

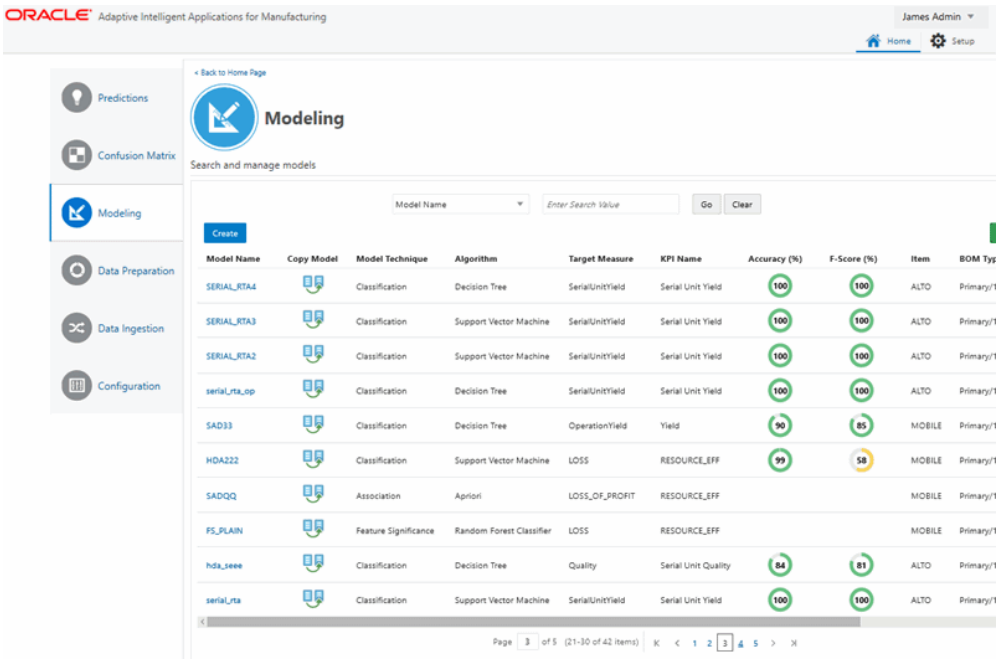
### To copy a model:

Alternatively, you can copy an existing model using the **Copy Model** icon. When copying a model, the original model's options are chosen by default, but you can change the model name, the training data split, the selected target and bins, the selected features, the algorithm, and deployment options from the original model's definition.

1. Navigate to the Model page.

   From the Home page, click **Predictions** or **Insights**, and then **Modeling**.

   The existing models appear in the Model page.




2. Click the Copy Model icon next to the model name that you want to copy.

3. Optionally, change the Model Name and Model Description.

   The model name defaults to the original model name, appended with "_1". For example, a copy of "model name" is named "model name_1".

4. On the Context page, for Prediction models only, optionally adjust the previously set training data split. Click Next.

5. The target from the original model displays on the Targets page. Optionally, add new bin ranges to the existing target or add a new target from the list of available targets.

6. Click Bins.

   Optionally, modify the bins for existing targets and define bins for new targets. You can also change the display names of selected targets and set turn classification bin alerts on or off.

7. Click Next.

8. On the Features page, optionally remove existing features, add new features, and change the feature display names.

9. Click Next.

10. On the Algorithm page, optionally change the selected algorithm's parameter

values or select another algorithm.

11. Click Next.

12. On the Deployment page, optionally change the original deployment options.

13. Click Next.

14. On the Summary page, review your selections for each model definition area. Click Previous to change your selections or click Submit.

### What's Next

Evaluating a Model, page 6-18

## Evaluating a Model

After building various models that use different parameters, input features, and algorithms or algorithm parameters to predict a specific target measure, you can evaluate each model to determine which model best fits historical data. Data scientists evaluate each functional type of model differently.

- Feature Significance - Evaluate how much each feature influences target measure results. Use the most significant features to create your Insights and Predictions models.

- Insights - Discover patterns in the historical work order data and publish the insights that influence the business outcome with a high level of factors influence.

- Predictions - Evaluate and review the performance of a predictions model using the confusion matrix. This matrix displays total predictions generated for the model and the percentage of true and false predictions by comparing predictions with actual results.

### To evaluate a feature significance model:

Use the Feature Significance page to determine which features in your feature significance model have a significant relationship with the model's target measure.

1. Navigate to the Feature Significance page.

   From the **Home** page, click **Insights**, then **Evaluation**, and then **Feature Significance**.

2. Click **View** in the **Significant Features** column for a model.

   The resulting page displays feature significance details for the selected model. This page displays:

- the target measure.
- a Significance Summary, which explains the number of significant features versus the number of features analyzed in comparison to a set of historical results data. If the model uses the Chi Square algorithm, there is also an explanation of how many significant features have a strong relation to the target measure.
- a Significant Features region, which provides a ranked and analyzed list of significant features.

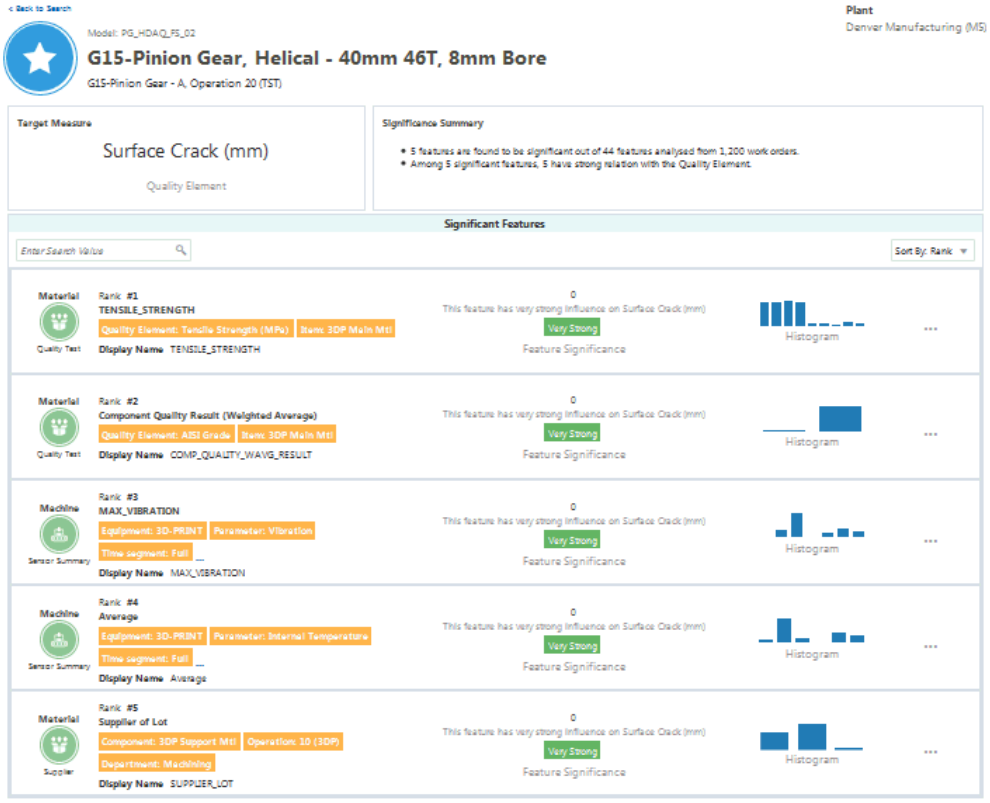
  **Tip:** Search for significant features using the Search field in the top left corner of the Significant Features region. The search criteria include:

  - Categories
  - Subcategories
  - Features
  - Entities
  - Relationships/Reasons

  Sort the significant features using the Sort By field in the top right corner of the Significant Features region. You can sort features by:

  - Rank
  - Category
  - Name

***View Significant Features page***



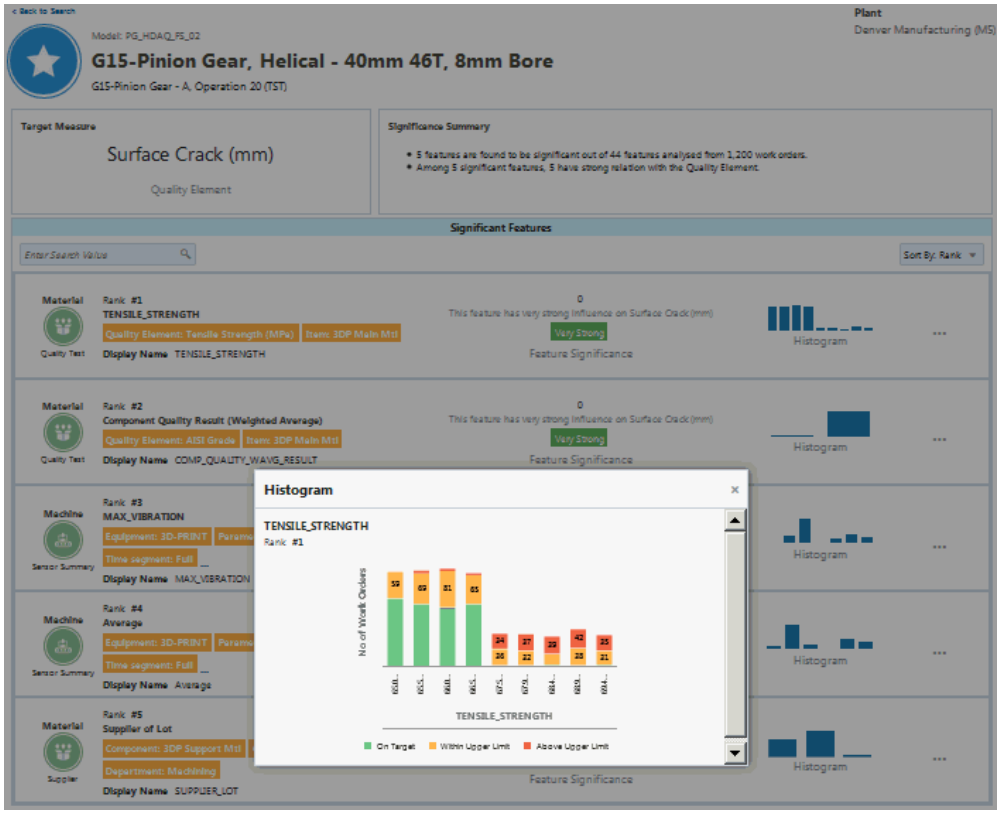

3. In the third column of the Significant Features region, view the following information about each feature:

   - If the model uses the Chi Square algorithm, a statement explains the strength of the feature's influence on the target measure. For example, in the screen shot shown above, the statement says "This feature has very strong influence on Surface Crack (mm)".

   - If the model uses the Random Forest Classifier algorithm, a number, ranging from 0 to 1, represents the importance of the feature's influence on the target measure. This number is displayed using a meter gauge format.

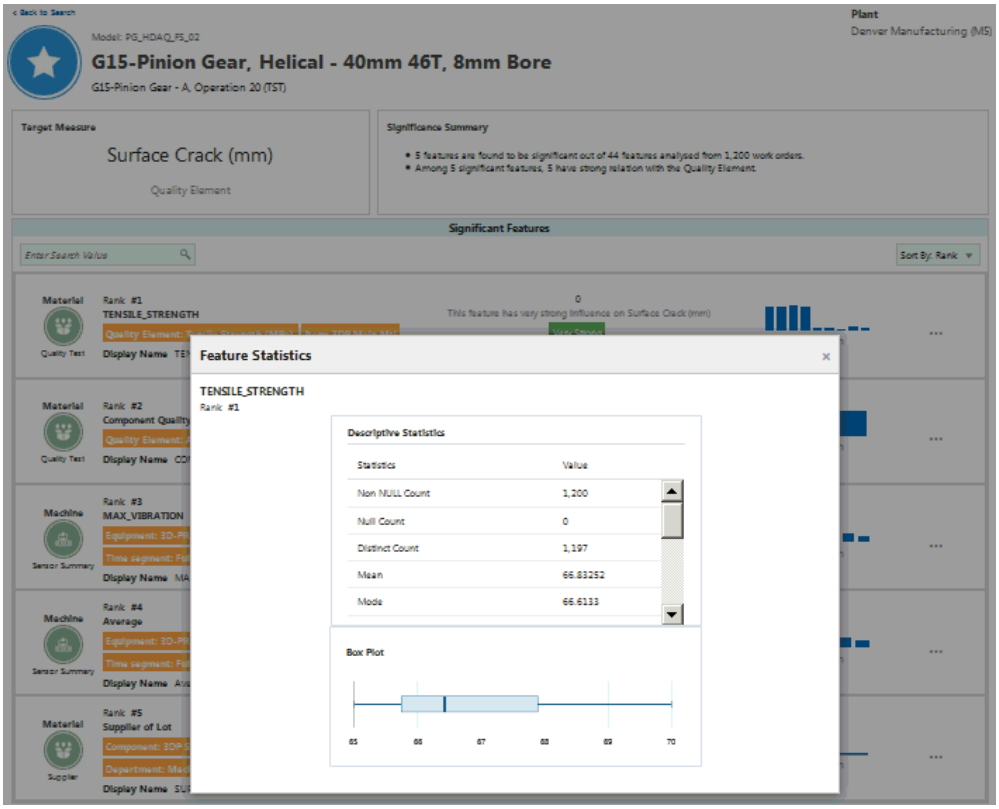
4. Click the bar chart for a feature to view a histogram showing how the feature's historical data falls within the model's classification ranges.

***Histogram***




5. Click the ellipsis points ( . . . ) on the right side of each significant feature listed to view the Feature Statistics page. This page includes the descriptive statistics of the feature and a box plot showing the data distribution for the selected feature.

***Feature Statistics***




## To evaluate an insights model:

1. Navigate to the Patterns & Correlations page.

   From the **Home** page, click **Insights**, then **Evaluation**, and then **Patterns & Correlations**.

2. Click **View** for a particular model name.

3. Review the results for the model:

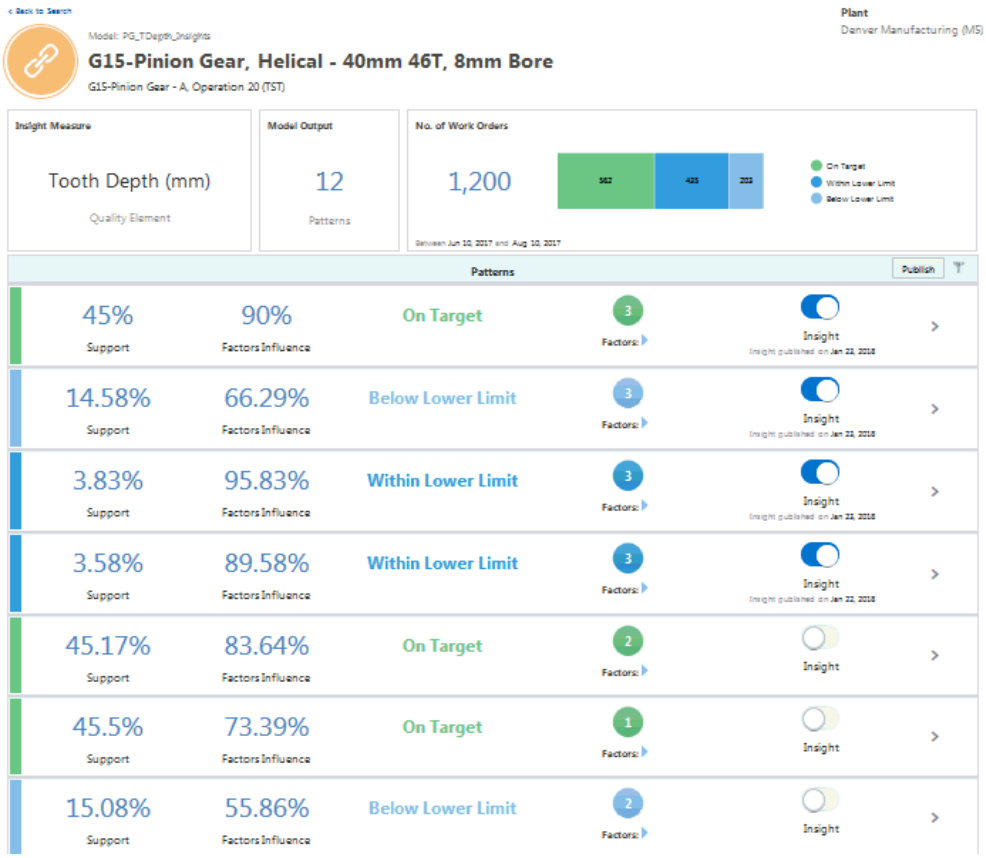
   - Insight Measure - the measure for which the model discovers influencing factors.

   - Model Output - the number of patterns (insights) found by the model. The model discovers various patterns/correlations between influencing factors and the target measure selected for the model.

   - Number of Work Orders - total number of historical work orders analyzed by the model. The chart provides a distribution of test results for the predefined

classification ranges. For serialized manufacturing, the number of serial units and the distribution of test results of serial units by predefined classification range displays.

***View Significant Features page***




4. Review the results for each insight.

   You can choose to publish up to ten insights for a given model. The top ten insights are selected for publishing by default. You can choose to modify the selections. Before publishing, review the support and factors influence percentage parameters for each pattern to determine which of them most influence the target measure. You must also apply your understanding of the business when determining which patterns to publish as insights to the end users in the Insights module pages. For each pattern, review:

   - Support - the percentage of historical work orders supporting the pattern out of the total number of work orders analyzed.

   - Factors Influence - the percentage of historical work orders that match the pattern out of all the historical work orders that match only the input factors.

- Factors - the number of factors which influenced the target measure to have a classification as discovered in the insight model.

5. Click the right arrow underneath the number of features to display the list of features.

6. Click the right arrow on the right side of the insight row to view the following charts:

   - Insight Measure Classification - displays the number of historical work orders where:

     - an insight pattern exists. The total number of work orders where the insight pattern exists (having both the input feature and the insight measure or the target measure matching as per the insight).

     - the input features match. The total number of work orders where only the input features match as per the insight but the target measure or the insight measure values are different from that of the insight..

     - a pattern does not exist. The total number of work orders where both the input feature and the target measure values do not match with the insight.

   - Correlation - correlation or distribution chart displaying the correlation (scatter plot) or distribution (bubble chart) between each influencing factor of the insight versus the target measure.

   - Insight Timeline - displays the target measure value information for all work orders matching the insight along a timeline. The target measure values are legends with the classification labels.

### To evaluate a predictions model:

1. Navigate to the Confusion Matrix page.

   From the **Home** page, click **Predictions** and then **Confusion Matrix**.

2. Select the model to evaluate.

3. Select Test Data, then click Refresh.

   The test data is a portion of the data set on which the model is tested. When the model is built, the data set is split into training data and test data. The model is created using the training data, but evaluated against the test data.

   In the example Confusion Matrix table shown below, the model generated:

   - Predictions: 78 predictions of the quality element SPEED from the test data.

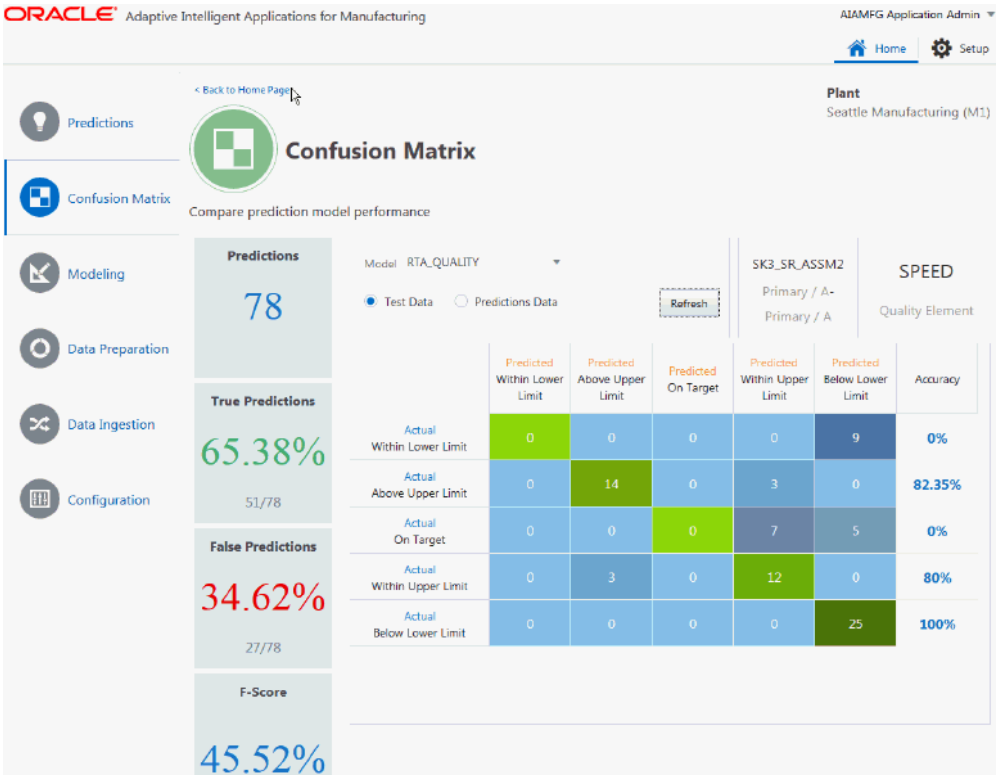
- True Predictions: 65.38% true predictions (percentage of predictions that matched actual results).
- False Predictions: 34.62% false predictions (percentage of predictions that did not match actual results).
- F-Score: 45.5%.

  The F-Score measures a model's accuracy, with a range between 0 - 100%. The higher the F-Score, the more accurate the model. The F-Score is computed as a harmonic mean between precision and recall. Use it as metric to evaluate model performance in cases of uneven class distribution or imbalanced data sets.

The matrix displays the predictions and compares them with actual results for each classification. The model shown below generated 12 Within Upper Limit SPEED predictions, but the SPEED of the assembly was actually within the upper limit 3 more times, for a total of 15 times. The accuracy of predictions are displayed by each classification.

> **Note:** You can also evaluate a predictions model using the most recent data once the actual work order results are available. To do this, run the program "Update Actuals for Prediction Model Targets" before evaluating the predictions model against the predictions data. For more information about this program, see Background Processes, page E-1.

***Example Confusion Matrix Table***



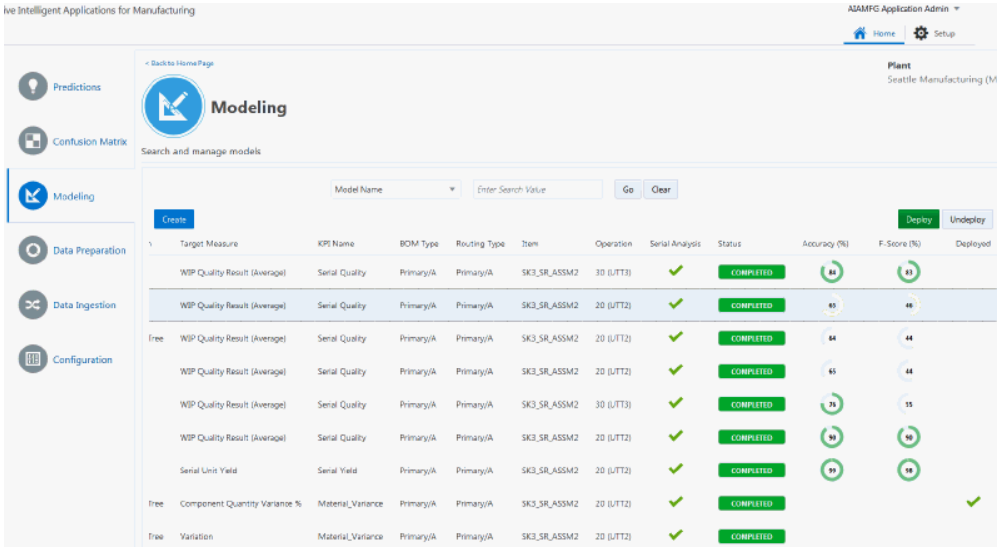

### To compare predictions models

4. Navigate to the Modeling page.

   From the **Home** page, click **Predictions** and then **Modeling**.

5. Search for the predictions models that you want to compare.

6. Evaluate the models based on their Accuracy (%) and F-Score (%).




## What's Next

Deploying a Model, page 6-27

# Deploying a Model

### To deploy a feature significance model:

Although you can create many feature significance models that use the same dataset and target measure, you can only deploy one of them at a time.

1. Navigate to the Feature Significance page.

   From the **Home** page, click **Insights**, then **Evaluation**, and then **Feature Significance**.

2. If multiple models for a certain context (which includes the dataset) and a target measure exist, decide which model to deploy.

3. Select the row for the model that you want to deploy.

4. Click **Deploy**.

   A green check mark appears in the Deployed column upon successful deployment.

   **Warning:** If you previously deployed another model with the same context and target measure, you must undeploy it before deploying another model. Select the row for the previously deployed model, then click **Undeploy**.

### To deploy an insights model:

Insights models have an auto-deploy option which is enabled for the initial insights model of a given dataset and target measure combination. Upon deployment of an insights model, the insights are published and appear on the Insights page. Each deployed model must contain at least one insight. Oracle recommends deploying an insights model only after evaluation. See: Evaluating a Model, page 6-18.

1. Navigate to the Patterns & Correlations page.

   From the **Home** page, click **Insights**, then **Evaluation**, and then **Patterns & Correlations**.

2. Select the row for the model that you want to deploy.

3. Click **Deploy**.

   A green check mark appears in the Deployed column upon successful deployment.

   > **Note:** You can deploy multiple Insights models with the same context and target measure. If you want to undeploy a model, select the row for the previously deployed model, then click **Undeploy**.

### To deploy a predictions model:

1. Navigate to the Modeling page.

   From the Home page, click **Predictions**, then **Modeling**.

2. Select a Predictions model row.

   > **Tip:** Select a Predictions model based on the model's accuracy and F-score.

3. Click **Deploy**.

   A green check mark appears in the Deployed column upon successful deployment.

   > **Additional Information:** If you previously deployed a model with the same context and target measure, you must undeploy it before deploying another model. Select the row for the previously deployed model, then click **Undeploy**.
   >
   > When you deploy a Predictions model, the Run Predictions Model background process starts and runs on the frequency specified in the Deployment option. Note that if a predictions model is

currently running for any in-progress work orders, you cannot undeploy the model. In such cases, use the Background Process page to cancel the Run Predictions Model program, and then undeploy the model. See Running Background Processes, page E-1 for information about how to cancel the predictions process or modify the predictions schedule.

# 7

# Presenting Data

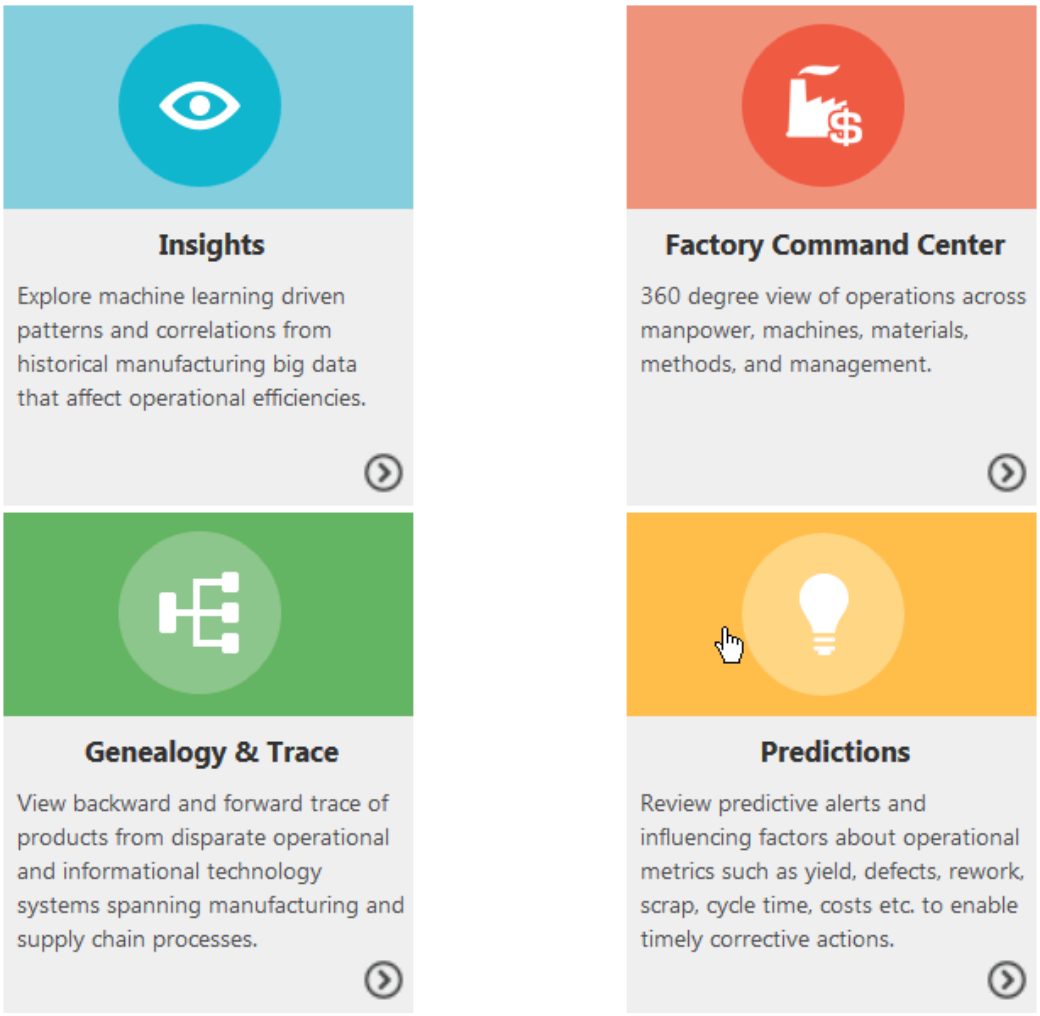
This chapter covers the following topics:

- Presenting Data Analysis Results
- Using Insights Analysis
- Using Predictive Analysis
- Using Genealogy and Trace
- Viewing the Factory Command Center

## Presenting Data Analysis Results

The Home page enables you to quickly navigate to the results of your data analysis of the past (Insights) or predictions of the future (Predictions). Alternatively, you can view every action on the shop floor using the Factory Command Center or view the linkages between actions and materials using Genealogy & Trace. This chapter explains the information you can gather from the data presented in these four modules.

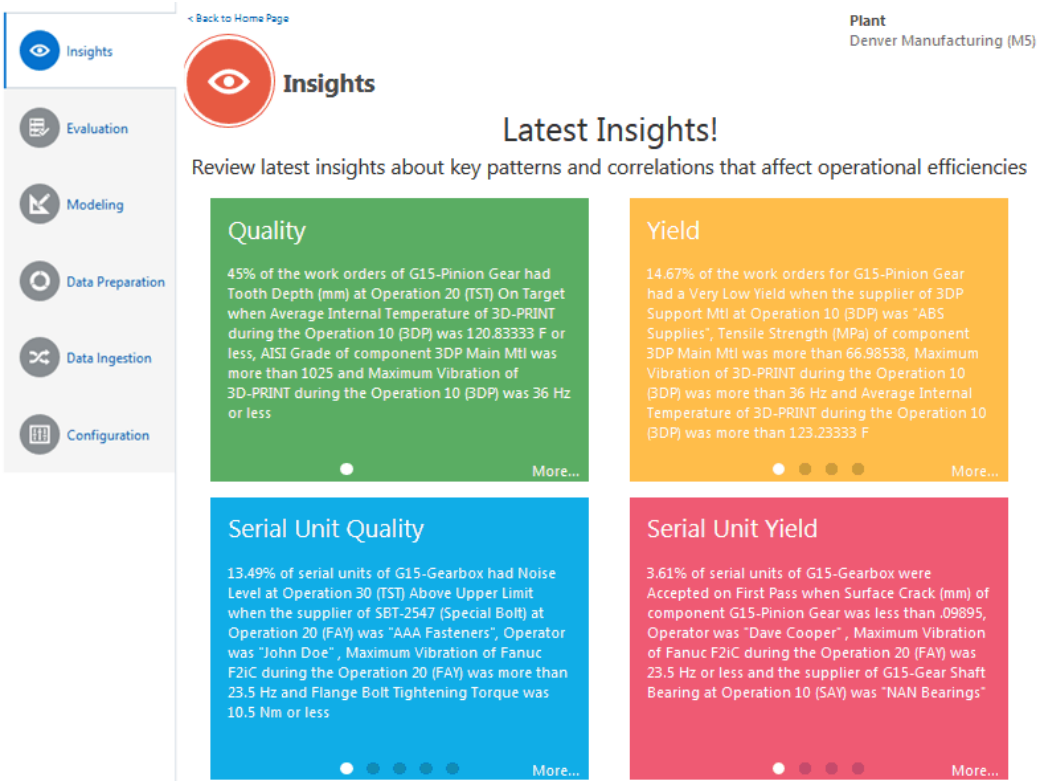
*Home page*




## Using Insights Analysis

The Insights page enables business users to explore published insights in order to understand the hidden factors influencing the output. Each tile on the Insights page summarizes the latest published insights for an output type (if there are no published insights for an output type, no tile displays). Output types include custom KPIs and the seeded KPIs of quality, yield, serial unit quality, and serial unit yield.

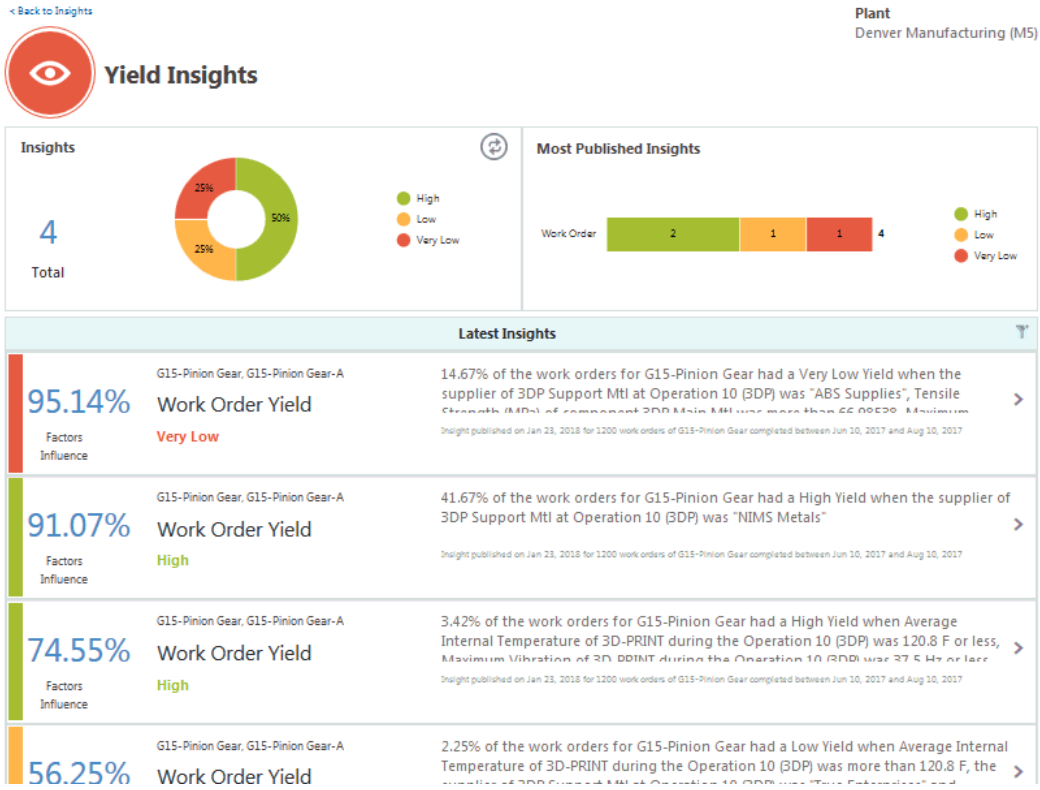
***Insights page***




Click the text for one of the top insights shown to navigate directly to the Insight (detail) page for that insight. Click the More link for one of the output types to view the last 90 days of published insights on an Insights (summary) page.

> **Tip:** Click the filter, then change the Published Date Range to view a time period other than the last 90 days.

***Yield Insights page***



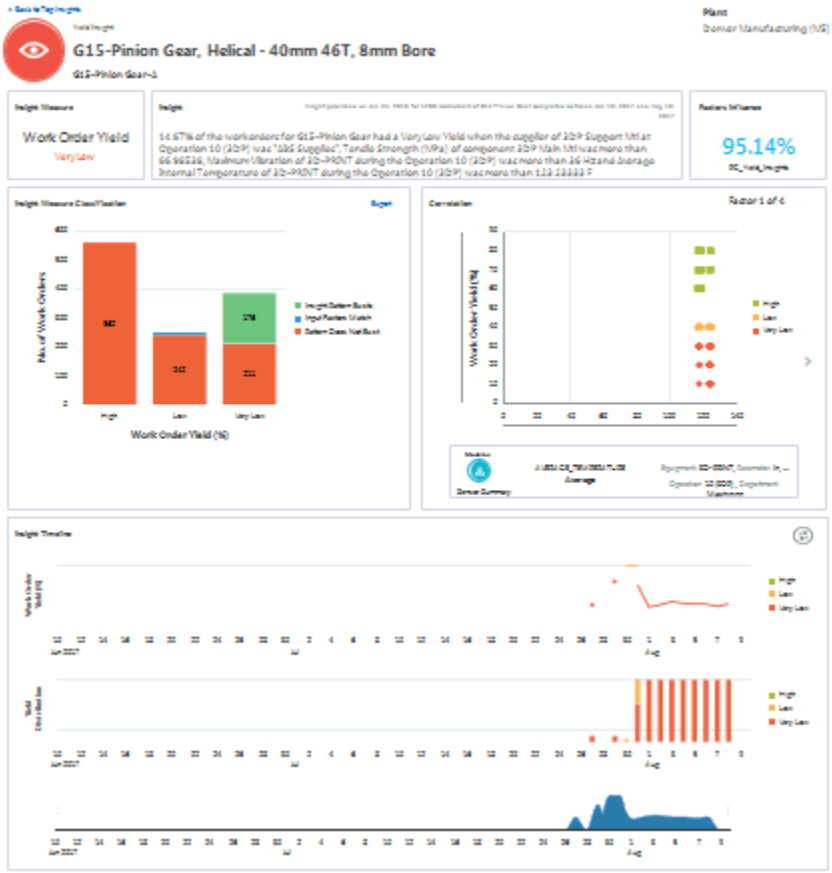

One insight model might result in multiple insights related to different output classifications. Depending on whether you are a process manufacturing, or discrete manufacturing user, you can search for insights using the following filter categories:

- Product Category
- Product
- Recipe/Work Order
- BOM Type/Revision
- Routing Type/Revision
- Operation
- Target Measure
- Insight
- Published Date Range (defaults to the last 90 days)

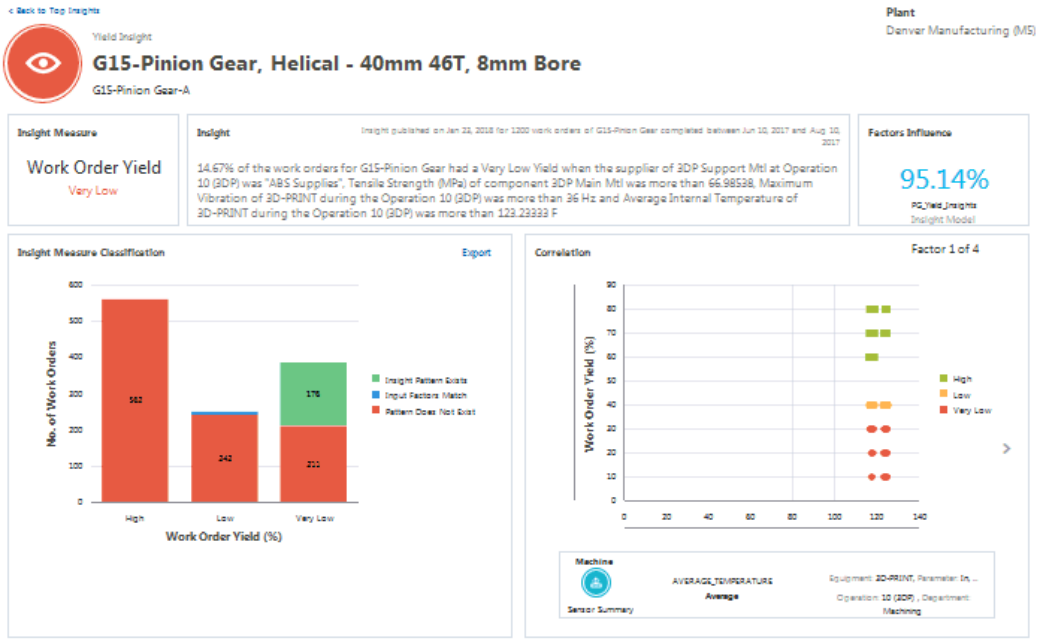
The insights are listed in the order of the published date. Users can drill into each of the

insights using the right arrow to understand the correlations and data distribution of the influencing factors for the output over the historical data analyzed. An example of an insight drill-down page is shown below as one page, and then displayed in two parts. This is the same page used when evaluating an insights model. See: Evaluating a Model, page 6-18.

***Yield Insight (detail) page***



***Yield Insight (detail) page, Part 1***




The first part of the above insight detail page displays the following:

- Insight Measure - a short description of the insight measure, in this case Work Order Yield, and the classification of this insight measure (Very Low).

- Insight - a written description that summarizes the insight pattern.

- Factors Influence - the percentage of historical work orders that match the pattern out of all the historical work orders that match only the input factors.

- Insight Measure Classification graph - shows the number of historical work orders where:

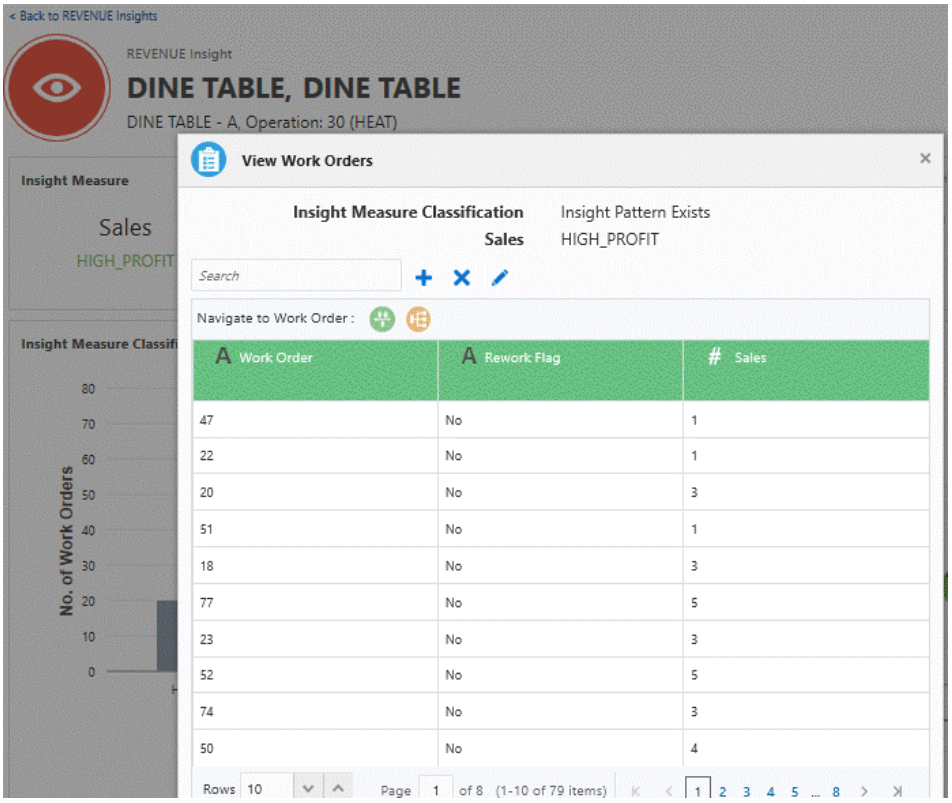
    - an insight pattern exists. The total number of work orders where the insight pattern exists (both the input factor and the insight measure or the target measure match, as per the insight). The color of the Insight Pattern Exists bar matches the insight measure defined bin color.

    - the input factors match. The total number of work orders where the input factors match as per the insight, but the target measure or the insight measure values are different from that of the insight.

    - a pattern does not exist. The total number of work orders where the input factor and the target measure values do not match with the insight.

**To view the history of a work order, serial number, or equipment instance:**

1. Click on the number of work orders (or serial numbers or equipment instances) in one of the above Insight Measure Classification graph categories to open the View Work Orders page.




2. Optionally, use the Search field to search for a particular work order or input factor.

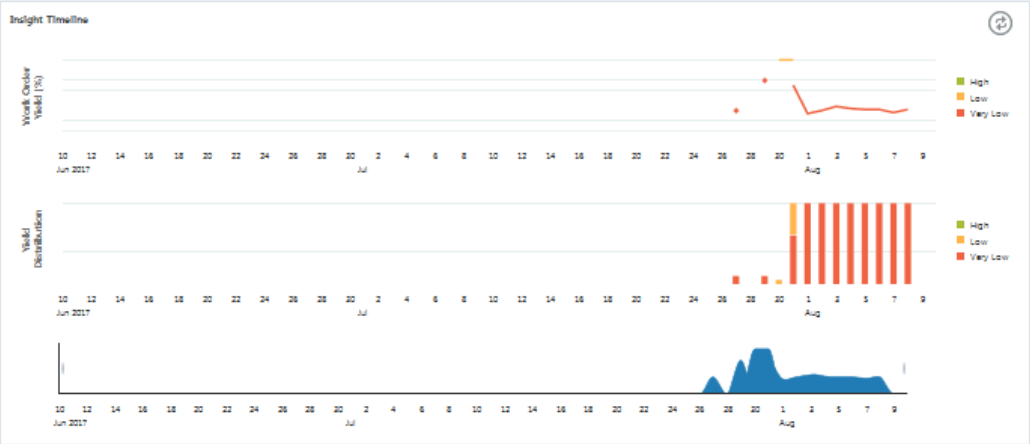
   Use the +, X, or Edit icons to add, delete, or edit search fields.

3. Select a work order (or serial number) in the list.

4. In the Navigate to Work Order (or Serial Number) region, select either the Timeline Viewer or Network Viewer icon.

   The Genealogy and Trace page opens. See: Using Genealogy and Trace, page 7-15

***Yield Insight (detail) page, Part 2***




The second part of the above insight detail page displays the:

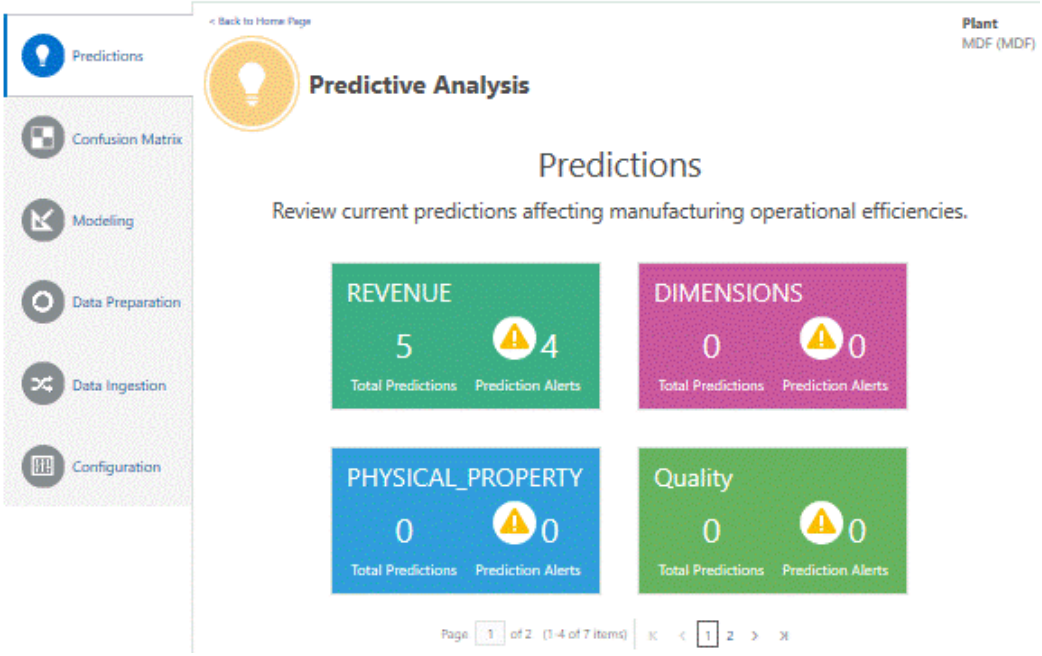
- Correlation graph, which shows the distribution (scatter plot) between each influencing factor of the insight versus the target measure.

- Insight Timeline, which shows the target measure value information for all work orders matching the insight along a timeline. The legend displays the target measure values with their classification labels.

## Using Predictive Analysis

Begin using predictive analysis by clicking Predictions on the Home page. The Predictive Analysis page appears and provides a starting point for reviewing the predictions across products in a plant, for a given product, and at the level of a work order and output (quality element, operation yield, work order yield, or custom KPI) combination. Next, select predictions for one of the seeded KPIs (Quality, Yield, Serial Unit Quality, or Serial Unit Yield), or predictions for a custom KPI.

***Predictive Analysis page***




Each Predictions page provides the following overview information for their type of prediction. Each type of Predictions page consists of two regions, both of which compile overview information for multiple predictions models, if more than one model is deployed. The two regions are:

- Predictions Summary.
- KPI Predictions, depending on the type of prediction selected.

The two regions are described below.

**Predictions Summary**

- Work Orders - the number of in process work orders for which the predictions models are running.
- Products - the number of products for which the predictions models are running.
- Predictive Models - the number of predictive models deployed and running.
- Serial Units (if applicable) - the number of individual serial units for which the predictions models are running.

**KPI Predictions**

- Total - the total number of predictions of this type for all products and all deployed models.

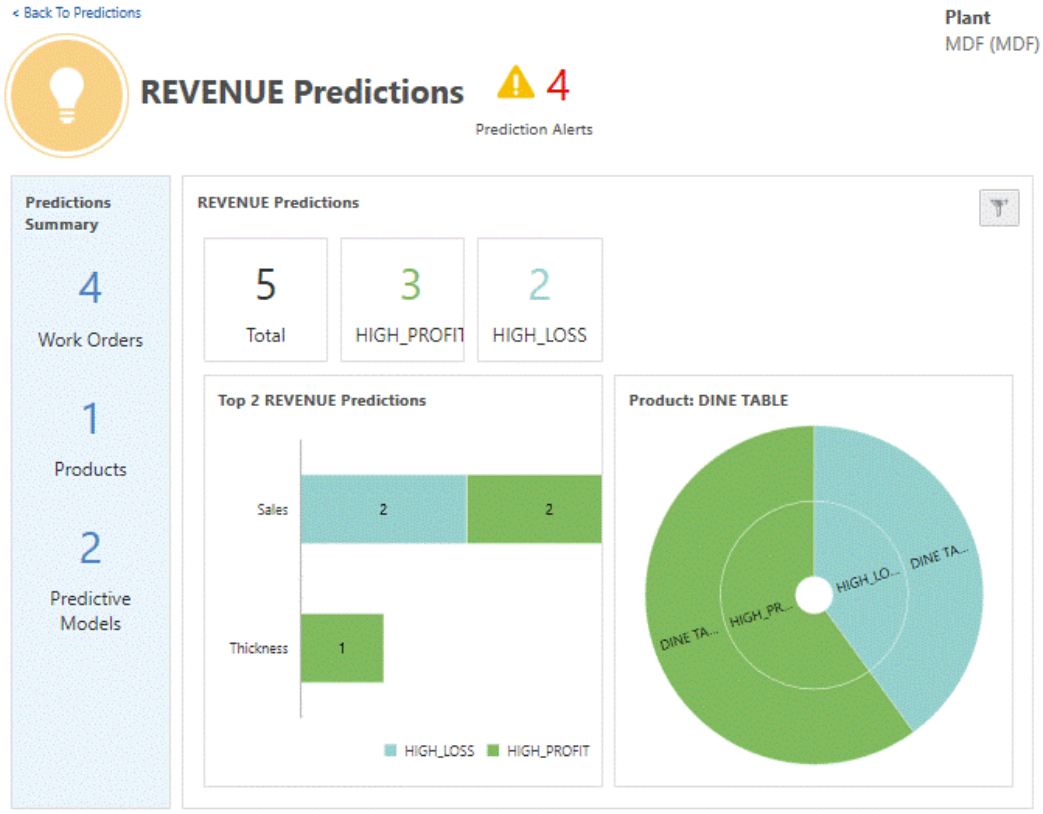
- Various classification predictions.
- Top KPI Predictions.
- Top Product Predictions.

  **Tip:** Use the Filter icon to view only certain predictions. Filter by:

  - Product category
  - Product
  - Operation
  - Target measure
  - Prediction (of a particular predefined classification label).
  - Status (of work orders). The Predictions page displays work orders with a status of In Process by default.

    View up to one month of completed or closed work orders.
  - Prediction with Alerts Only

***Example of a KPI Predictions page***




In the KPI Predictions region, click a prediction number or chart point to view a list of the specific work orders or serial units predicted.

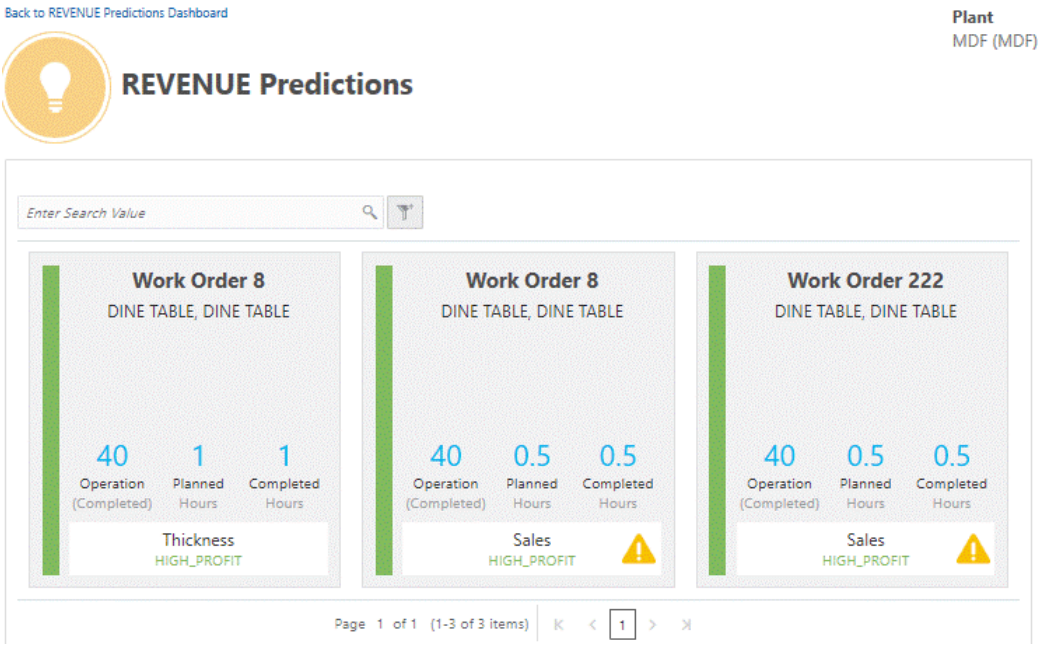
**Tip:** Use the Filter icon to view only certain work orders or serial units. Filter by:

- Product category
- Product
- Operation
- Target measure
- Prediction (of a particular predefined classification label).
- Status (of work orders). The Predictions page displays work orders with a status of In Progess by default.

  View up to one month of completed or closed work orders.

- Prediction with Alerts Only

***Revenue Predictions of High_Profit***




**Additional Information:** Predictions for work orders from case record data only specify whether the work order operation is In Progress or Completed.

Click on an individual work order or serial unit to view the prediction details. Prediction details include:

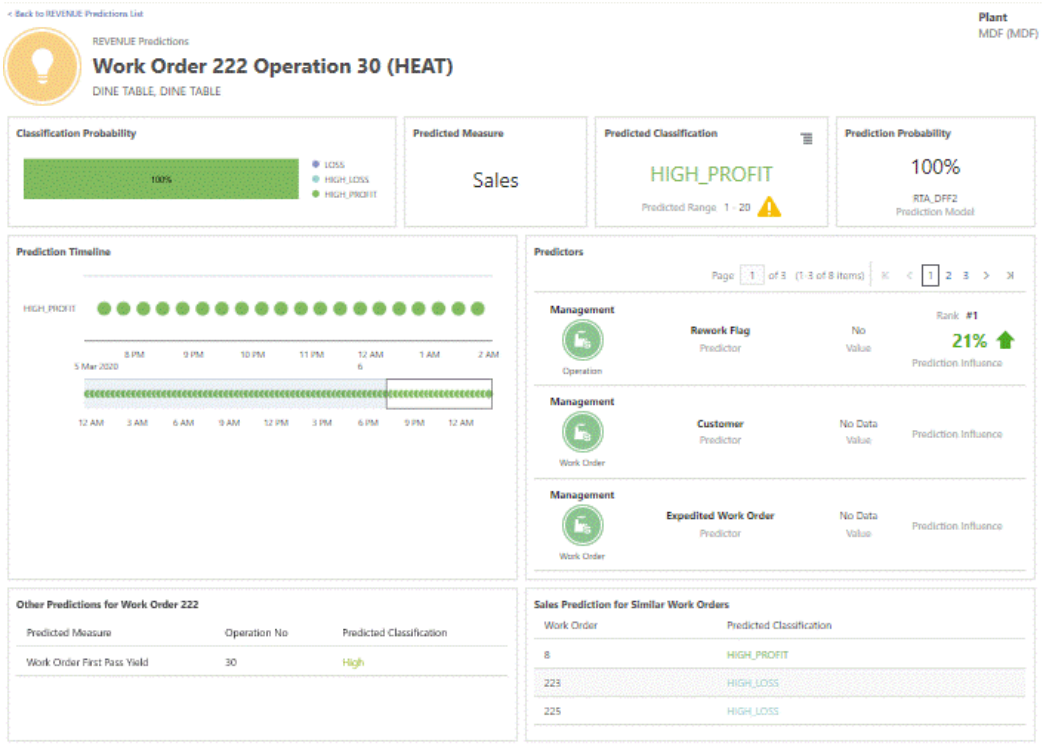
- Classification Probability - The probability of each classification occurring for this work order.
- Predicted Measure - The predicted measure displays the display name of the target element used in the prediction model.
- Predicted Classification - Displays the current predicted classification. Click the Actions icon to display a pop-up list of all prediction classifications.
- Prediction Probability - The probability of the predicted measure falling within the predicted classification.
- Prediction Timeline - Displays the prediction values over a selected time range.

- Predictors- Displays the percentage of the influence that each predictor had on the probability of the predicted measure. It can be positive or negative.

- Other Predictions - Displays the predictions for other predicted measures if there are any models built with the same context operation but with a different predicted measure.

- Prediction for Similar Work Orders (or Serial Units)- View other Pending or In progress work orders or serial units to see how the same model predicts the same predicted measure for the other work orders or serial units.

***Work Order Prediction Detail page***




## Viewing Predictor Details

From the work order or serial number prediction details, select a predictor to access the Predictor Details page. This page explains the distribution of the predictor data from the training data set and the related statistics. Regions on the page include:

- Prediction - displays the predicted classification and the related context for the predicted output. The context includes output, the product item, work order or serial unit, operation, and department.

- Predictor - displays the predictor display name and context. The context for the

predictor includes the category, subcategory of the predictor, and the parameters related to the context. For example, for the resource usage, the context parameters include resource information at the first level, then operation and activity information at the second level.

- Value - displays the absolute value for the predictor for the currently running work order or serial unit, the range this predictor falls within the model, the prediction influence, and rank.

- Historical Dataset - Refers to the data set that matches the context criteria. The model building process excludes records with no output.

    - Predictor Details - The feature significance model supplies the descriptive statistics of the predictor. Therefore, predictor details only display if the predictor is chosen as the significant feature through a deployed feature significance model. The descriptive statistics like min, max, mean, mode, quartile ranges, and so on, are displayed for the predictor. For more information about the descriptive statistics, see: *To evaluate a feature significance model* in Evaluating a Model, page 6-18.

    - Predictor Histogram - Displays the total number of work orders or serial units for each predictor value range. A gray box highlights the range which contains the current predictor value.

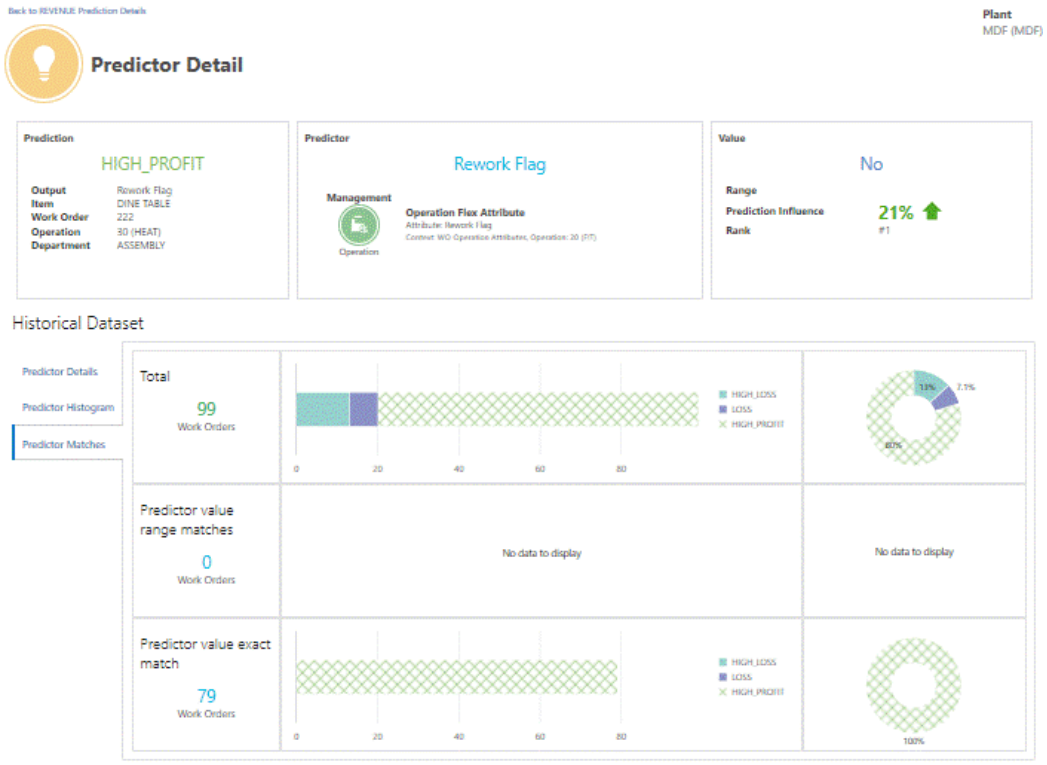
    - Predictor Matches - Displays the distribution of the predictor data by prediction classification. The information displayed includes:

        - The total number of work orders or serial units.

        - The number of work orders or serial units that fall within a predictor range.

        - The number of work orders or serial units that match a predictor absolute value.

***Predictor Detail page***




## Using Genealogy and Trace

The Genealogy and Trace page enables you to navigate through the entire history of a work order, serial number, equipment instance, or lot using a Timeline Viewer. Events that appear in the timeline in relation to an entity include Enterprise Resource Planning (ERP) events related to work orders, purchase orders, sales orders and other logistical movements of entities, and events captured by sensor devices.

Genealogy and Trace also includes a Network Viewer, which traces the history of item lots and serial units from their inception to their current location within and across organizations. Every action performed on the entities through their journey is mapped. Users can drill down into the details of a relationship between two nodes in a genealogy, details of participating entities, and genealogies of all the participating entities. You can launch a genealogy by providing a purchase order, lot, serial number, work order, sales order, or a user defined entity. Using the two viewers together enables users to quickly isolate the event or material that resulted in a quality issue and/or a yield loss in a finished product.

You can extend the timeline and network viewer by adding new types of events or entities using REST services. Customize a timeline event in the following ways:

1. Choose the type of entity for which the event occurs, the entity name, and the

manufacturing plant (for example, Work Order, Work Order Number and Organization Code).

2. Specify text that describes the event.

3. Specify the date on which the event occurs.

4. Choose an icon and its background color to visually represent the new event. Choose from the icons listed in Icon Mapping, page D-100.

5. Choose the text font and color to use when displaying the event node.

6. Specify the details of the event to display when a user right clicks on an event node and selects Show Details.

Customize a node or entity in the Network Viewer by:

1. Choosing the type of entity that the node represents and its code (for example, Lot, Work Order, Lot Number and Work Order Number).

2. Choosing the inventory item associated with the entity.

3. Entering text to describe the node.

4. Selecting an icon for the node (only required if the entity is user defined).

5. Choosing the text font and color to use when displaying the network node.

6. Specifying the details of the entity to display when a user right clicks on a network node and selects Show Details.

7. Adding user defined indicator icons for use in Network Viewer entity nodes. Choose from the indicator icons listed in Icon Mapping, page D-100.

    - Use object types above 100 for user defined entities.

    - Review the following two inbound web service sets in Overview of Oracle Adaptive Intelligent Apps for Manufacturing REST Web Services, page D-1 to understand how you can customize an event:

        - Custom events for Timeline Viewer.

        - Custom nodes for Network Viewer.

### To use the Timeline Viewer:

The Genealogy and Trace page presents the timeline history in a split screen, with machine and custom event data displayed in the bottom half of the screen. Events related to a work order, equipment, serial number, or lot are captured directly by machines and display in the bottom half. ERP events display in the top half. Different icons represent each event type. See: Icon Mapping, page D-100.

1. Navigate to the Genealogy and Trace page. Click the Search icon. Select Timeline.

2. In the Select Entity field, select Work Order, Equipment, Serial, or Lot.

   A search criteria region appears.

3. Enter your search criteria, then click Search.

4. Select an entity from the search results.

   A timeline view of all events associated with the entity appears. The timeline extends over the entire period of events. For example, if all of the events in a work order occur in a 4 hour window, the timeline extends to this entire duration, with scaling in hours. For equipment, the timeline initially displays one week, but you can change the start and finish dates at the top of the timeline as needed. A lot or serial timeline, like a work order timeline, displays the entire history of the serial unit or lot.


5. Optionally, adjust the timeline to zoom in or out on a time period. Perform this adjustment by moving the left and right ends of the lower region timeline slider. You can also drag your view of events to the left or right.

**Additional Information:** The timeline slider uses boxes to represent the concentration of events along the timeline. In the upper section of the timeline, scroll up, down, left, and right to view all events, if necessary.

6. Optionally, select an event or entity, right click, then select Show Details.

    - For all entities, you can right click to View Network or View Serial Timeline, if the entity selected is a serial assembly.

    - For a serial entity, you can also right click to View Serial Timeline. The timeline opens to the same point in time, but only displays events relevant to the serial number.

    - For an equipment entity, you can also right click to View Equipment Timeline. The timeline opens to the same point in time, but only displays operations and events relevant to the equipment. If multiple equipment instances exist, all instances appear above the timeline. Select one of the instances to view in the timeline.

    - For equipment entities, operations appear in the top half of the timeline, while events and alerts appear in the lower half.

    - For a lot entity, you can also right click to View Lot Timeline from the inception to the consumption of the lot. Examples of lot timeline events include receipt to stores, RMA return, miscellaneous receipt/issue, WIP issue, WIP completion, and lot translate.

    - Events do not appear on the timeline if the events occur in an organization for which you do not have access.

    - If more than one event occurs at the same time, the events appear stacked. The number of stacked events displays in the lower right corner of the top event. Click this number to expand the view of all events.

7. Optionally, narrow the list of entities and events shown on the timeline by selecting the operations or types of events to display at the top of the timeline.

    - For work orders, select the operations and types of events.

    - For equipment, serial, and lot numbers, select the types of events.

### To use the Network Viewer:

Network Viewer enables you to trace a serialized or lot controlled item from a supplier, through its consumption in a manufacturing work order, to its shipment to a customer. Access the Network Viewer from a selected entity in the Timeline Viewer or directly from the Genealogy and Trace page.

> **Warning:** The Network Viewer appears blank when viewed with Microsoft Internet Explorer. Use another supported browser. See: Supported Browsers, *Oracle Adaptive Intelligent Apps for Manufacturing Data Ingestion User's Guide*.

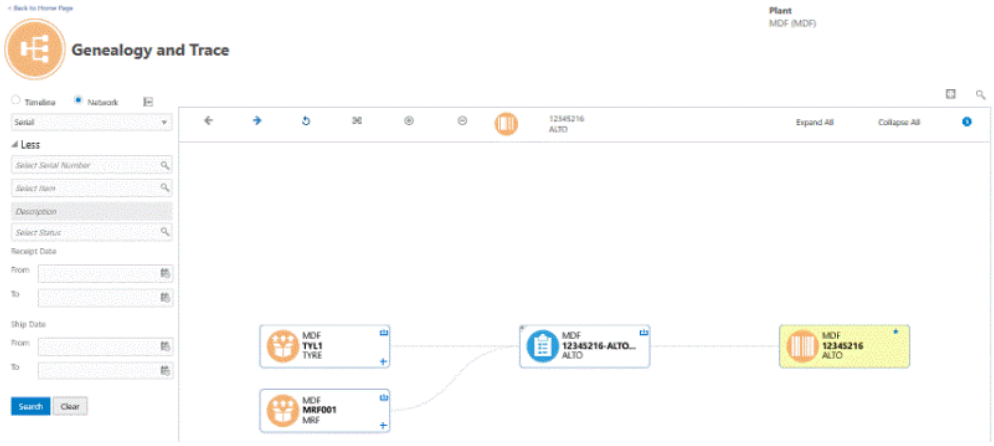
1. Navigate to the Genealogy and Trace page. Click the Search icon. Select Network.
2. In the Select Entity field, select Lot, Serial, Work Order, Purchase Order, Sales Order, or User Defined.

   A search criteria region appears.
3. Enter your search criteria, then click Search.

   > **Note:** For entities with a large list of values, you can not view any values unless you narrow the list of values by entering search criteria.
4. Select an entity from the search results.

   Once the genealogy of an entity appears, the entity itself is highlighted yellow. You can navigate through the genealogy and drill into the details of all participating entities. Whenever you perform a navigation action on a different entity, that entity is identified by a star in the top right corner. So, at any given time, you can identify the entity at the current center of analysis (the entity with the star) and the original entity (the entity highlighted yellow).

   > **Additional Information:** A red work order indicates scrap produced.




5. Optionally, right click on an entity and select one of the following options:

   **Note:** Options vary depending on the type of entity selected.

   - Show Details - displays entity header information, including alerts.
   - View Work Order, Equipment, Lot, or Serial Timeline - displays the Timeline Viewer.
   - Show Parent and Child Nodes Only - shows all entities directly related to the entity with a star.
   - Highlight Parent and Child Nodes - displays all nodes, but indirectly related nodes are grayed out.
   - Show Adjacent Nodes Only- displays only the entities immediately before and after the entity with a star.
   - Unhide Adjacent Nodes - reveals nodes adjacent to visible nodes when the hidden, adjacent nodes were either not part of the original genealogy retrieved or were hidden when an option such as "Show Parent and Child Nodes only" was used.
   - Show Entire Network - comprehensively explodes through the entire supply chain, showing all nodes connected to the selected node.

6. Notice the following types of lines that connect the entities in a network:
   - Solid black - the entities have a direct relationship to each other. For example, a solid black line connecting a purchase order to multiple serial numbers indicates that the serial numbers were purchased with that purchase order.

- Dotted - the entities have some sort of relationship, but you must further expand the entities shown to understand the relationship. For example, a serial number may have a relationship with a work order, but you can only see that a parent serial number was used to make a child serialized assembly once you expand the work order.

- Solid red - indicates an item return.

  **Additional Information:** A link to a lock icon labeled "No Access" indicates a transfer to an organization that is inaccessible to the user.

7. Optionally, right click on a connecting line to view the transaction details.

8. To display the object type of an entity, hover over the entity's icon.

9. Optionally, expand entities with a plus sign next to them. An expanded purchase order, for example, might display multiple lots. An expanded lot might display all of the serial items contained in the lot.

   - Use the Expand All and Collapse All links in the upper right corner of the page to expand or collapse all entities at once.

   - Use the Control Panel icon in the upper left corner of the page to zoom in, out, or to fit the view to the page.

   - Use the Backward, Forward, and Refresh icons in the upper left corner of the page to view previous or next pages and to reset the view to where you started.

   - A serial number displayed as a stack of cards indicates that multiple serial numbers of the same item have a relationship with the selected entity.

   **Note:** If you are using Business Entity Data, the information you can view for process manufacturing and discrete manufacturing organizations in the Genealogy and Trace module depends on the Business Entity Data ingestion method.

   The following table displays the information associated to the Genealogy and Trace module that is supported and not supported for both process manufacturing and discrete manufacturing organizations, depending on the Business Entity Data ingestion method you use:

| Genealogy and Trace | Information | Support Using CSV file Upload | Support Using REST Web Services Upload |
|---|---|---|---|
| Network | Node Details (Some of the Details attributes) | No | No |
| | Internal Orders | No | Yes |
| Equipment Timeline | Maintenance Events | No | No |
| | Serial Timeline (Some of the Details attributes like Subinventory and so on.) | No | No |
| | Subinventory Transfers | No | Yes |

## Viewing the Factory Command Center

The Factory Command Center displays a real-time overview of factory events, categorized into the 5 Ms of the factory: manpower, machines, materials, management, and methods, which are all linked to shop floor work orders. These are the same categories by which predictors are grouped in the Insights and Predictions modules and there is a continuous correlation between the current factory status and predictions of future events at any given time.

The Factory Command Center enables you to view at a glance the numerical information for the seeded key performance indicators (KPIs) listed under each category of Manpower, Machine, Management, Material, and Method on a single page. The numerical values are color coded to indicate a positive (green), neutral (blue) or negative (red) value. You can customize KPIs in each category as described below using REST web services:

- Add custom KPIs.
- Reorder KPIs according to priority and relevance.

- Hide KPIs that hold no significance to your organization.

See REST Web Services, page D-1 for more information.

From the Factory Command Center, click **Manpower**, **Machine**, **Management**, **Material**, or **Method** to view details of each of the 5 Ms of the factory.

Apart from showing the key performance indicators (KPIs), the Factory Command Center also displays the total number of prediction alerts that exist for the plant at the top. The information on this page is shown in the context of the specific plant (and, potentially, the department) selected in the User Preferences.

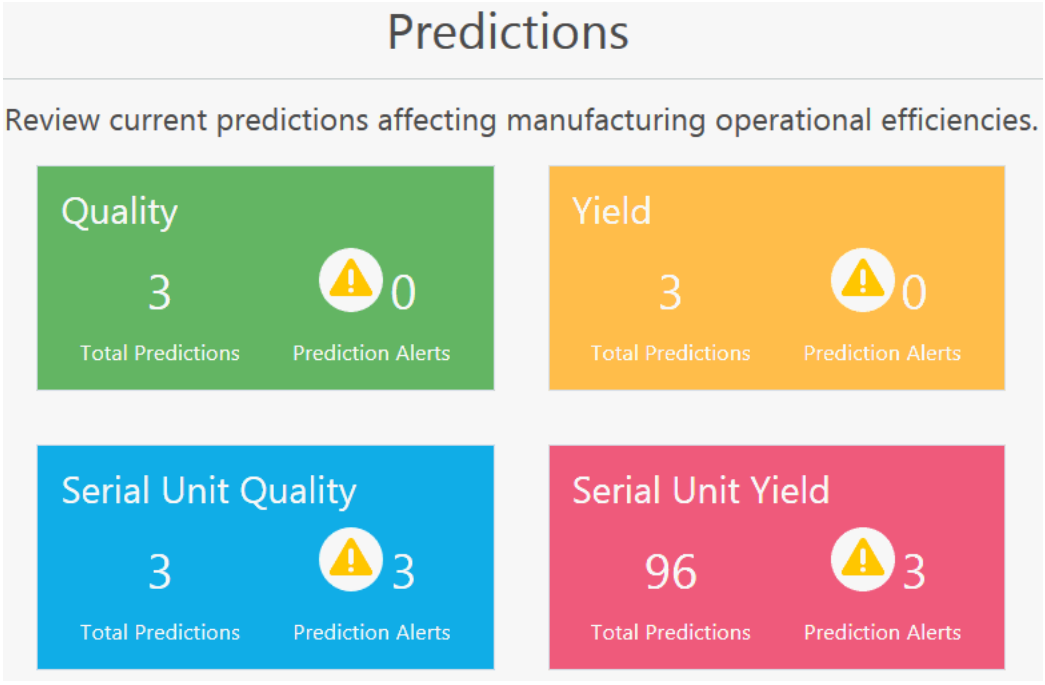
***Factory Command Center***


## Viewing Predictions

The number of Prediction alerts that display at the top of the Factory Command Center indicate the total number for the plant/organization selected on the User Preferences page. See Setting Up User Preferences, page 2-15 for more information.

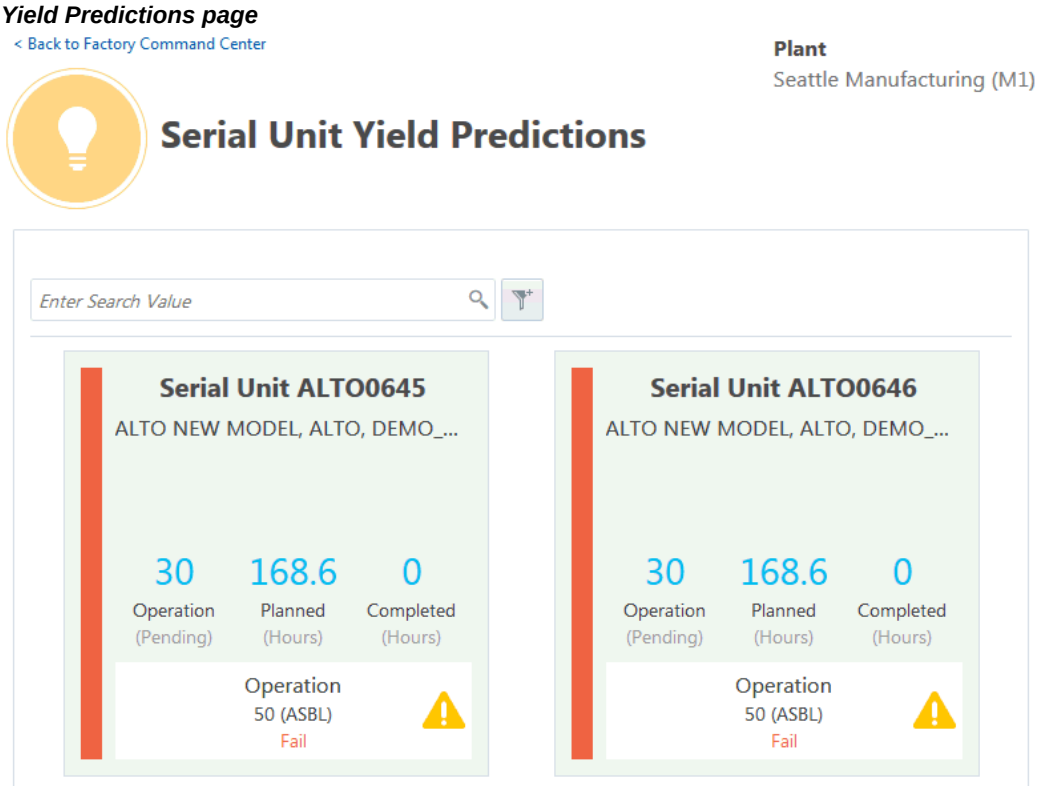
From the Factory Command Center, click **Prediction Alerts** that appears at the top of the page. The Predictions page appears, which groups the alerts into the seeded and user defined Key Performance Indicators (KPIs) according to the display sequence setup from the KPI definition.

***Predictions page***




At this point, you have left the Factory Command Center and navigated to the Predictions module. See Using Predictive Analysis, page 7-8 for more information about the Predictions module. You can click the prediction alerts for each of the above categories to see details, but, from the Serial Unit Yield Predictions page, for example, you can also click the Back to Factory Command Center link to return to monitoring the factory status.

***Yield Predictions page***




## Viewing Manpower

The Factory Command Center displays the numerical values for the following KPIs listed under the category of Manpower. These KPIs enable you to monitor personnel on the shopfloor:

- With Prediction Alert: Number of operators who are currently assigned or clocked in for work orders which have prediction alerts.
- Total: Number of operators in the plant or department, if specified, for the current work shifts.
- Not Reported: Number of operators who have not reported for work shifts.
- Job Unassigned: Number of operators who have not been assigned or associated with any of the currently running open work orders
- Not Clocked in: Number of Operators who are not clocked-in to any of the currently running open work orders.
- Skills Mismatch: Number of clocked-in operators of the currently running open work orders whose skill set defined in the human resource system does not match

the required skills for the job operation defined in the production execution system.

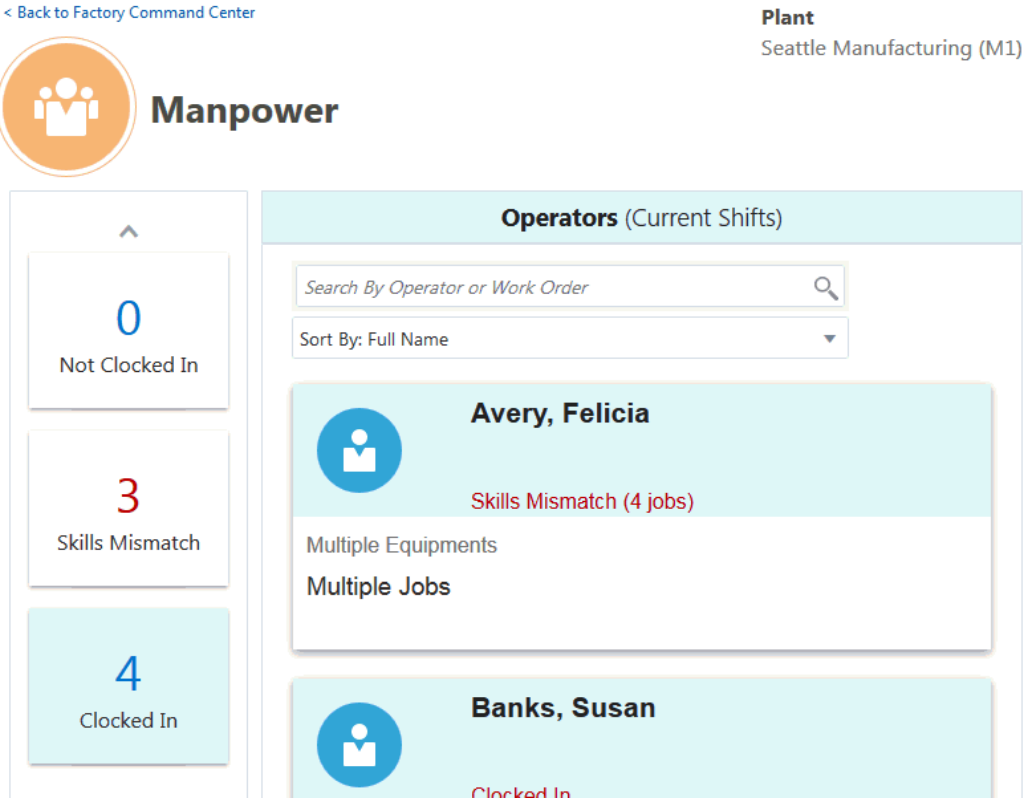
- Clocked in: Number of operators who are clocked-in to any of the currently running open work orders.

Click any of the above KPIs to filter the list of operators working by KPI in the Manpower page. To filter by another KPI, click a KPI listed on the left side of the Manpower page.

> **Tip:** You can also sort information using the following Sort By options: Full Name or Step completion date.

The Manpower page displays the last and first name of the operator, the option to email the operator, and the operator status. The equipment description or equipment number and job description or number of jobs for which they are clocked in also displays.

***Manpower page***




## Viewing Machines

The Factory Command Center page displays the numerical values for the following Machine KPIs, which enable you to monitor equipment on the shopfloor:

- Down: Lists the equipment currently down or not providing event information.
- Job Unassigned: Equipment without a work order operation assigned.
- Under Maintenance: Equipment with open scheduled maintenance work requests.
- Alarms & Events: Equipment with alarms, such as vibration spike, temperature spike, humidity spike or other events, such as machine door open/closed, machine power up/power down, machine up/down, or job started/job paused/job resumed/job completed within the configured display window time.

  **Additional Information:** You can specify the last number of hours over which to show the alarms that occurred by entering a value in the User Preferences page, Alarms last N hours field. See Defining User Preferences, page 2-15 for more information.

- In Use: Equipment currently running and providing event information.
- Total: Total equipment and equipment instances within the plant and selected department, if any.
- With Prediction Alert: Lists equipment with currently assigned work order operations that have prediction alerts.

Click any one of the KPIs to view details in the Machine page. To view details of another KPI, click a KPI listed on the left side of the Machine page.

**Tip:** You can sort information for a KPI in the Machine page using the following Sort By values:

- Item
- Work Order
- Equipment
- Completion date

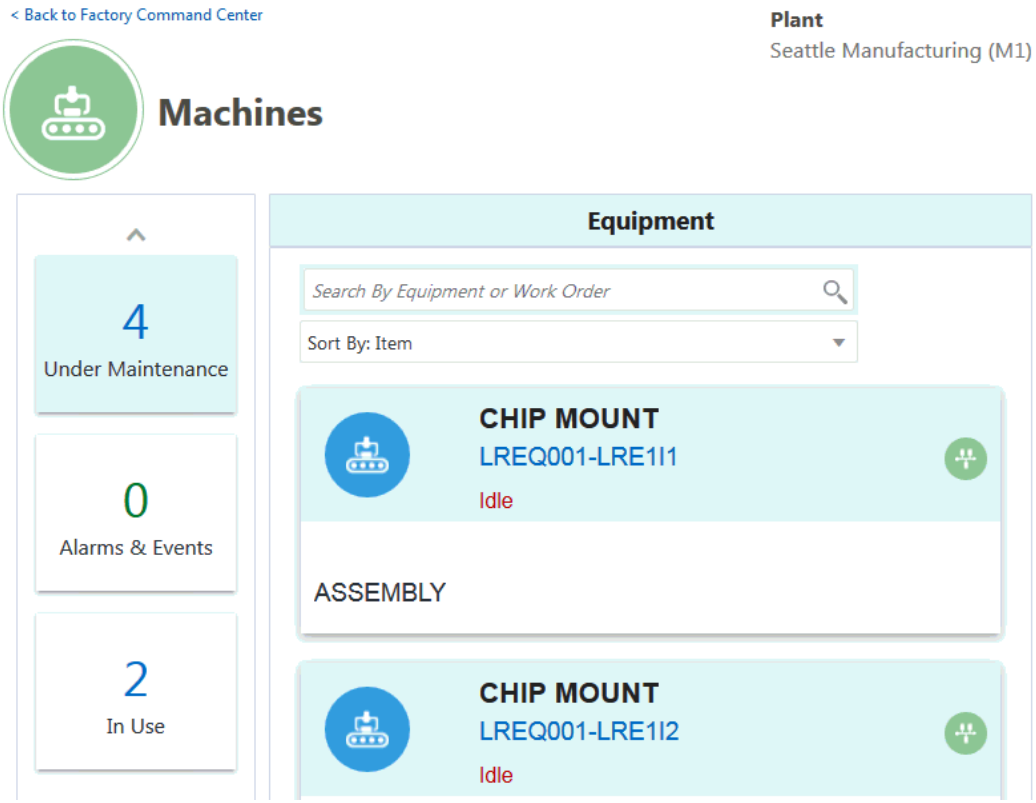
Filter information by equipment or work order using the Search By field.

The Machine page displays each equipment instance as an individual tile. The header of each tile includes the machine/resource name and instance name. Other possible tile information includes:

- Alarm/event

- Equipment instance status (up, down, in use, etc.)
- In process product currently in production

***Machines page***




## Viewing Management

The Factory Command Center displays the numerical values for the KPIs listed under the Management category, which enable you to monitor work orders currently in process on the shopfloor. Additionally, you have the option to include work orders that are:

- due to start soon.
- delayed beyond a certain number of hours.
- recently completed.

Include future, delayed, and completed work orders by setting the following parameters in the User Preferences page. See Setting Up User Preferences, page 2-15 for more information.

- Pending Work Order Time Fence (days): Display all work orders scheduled to start within the number of days specified in this field.
- Delay Tolerance (hours): Display all work orders delayed beyond their scheduled start time plus the delay tolerance specified in this field.
- Completed Work Order Time Fence (days): Display all work orders completed within the past number of days specified in this field.

The Factory Command Center displays the following KPIs:

- Start Delayed: Number of work orders not started that have a scheduled start date/time in the past and greater than the delay tolerance.
- Operation Delay: Number of work orders with any operation not started that have a scheduled completion date/time in the past and greater than the delay tolerance.
- Finish Delayed: Number of work orders with a scheduled completion date/time in the past and greater than the delay tolerance.
- On Hold: Number of work orders with the status On Hold.
- Ship Date Overdue: Number of work orders with a reserved sales order ship date/time in the past and greater than the delay tolerance.
- Pending Inspection: For Discrete Manufacturing, this includes the number of work orders with any operation that has a shop floor status of Pending Inspection. For Process Manufacturing, this includes the number of samples with a status of Pending Testing.
- Quality Issues: Number of currently running open and recently completed work orders with rejected and/or scrap quantities.
- Total: Number of currently running open work orders.
- With Prediction Alerts: Number of currently running open work orders that have prediction alerts.

Click one of the KPI to view details in the Management page. To view details of another KPI, click a KPI listed on the left side of the Management page.

> **Tip:** You can sort information for a KPI in the Management page using the following Sort By values:
>
> - Work Order
> - Due Date

- Product
- Customer

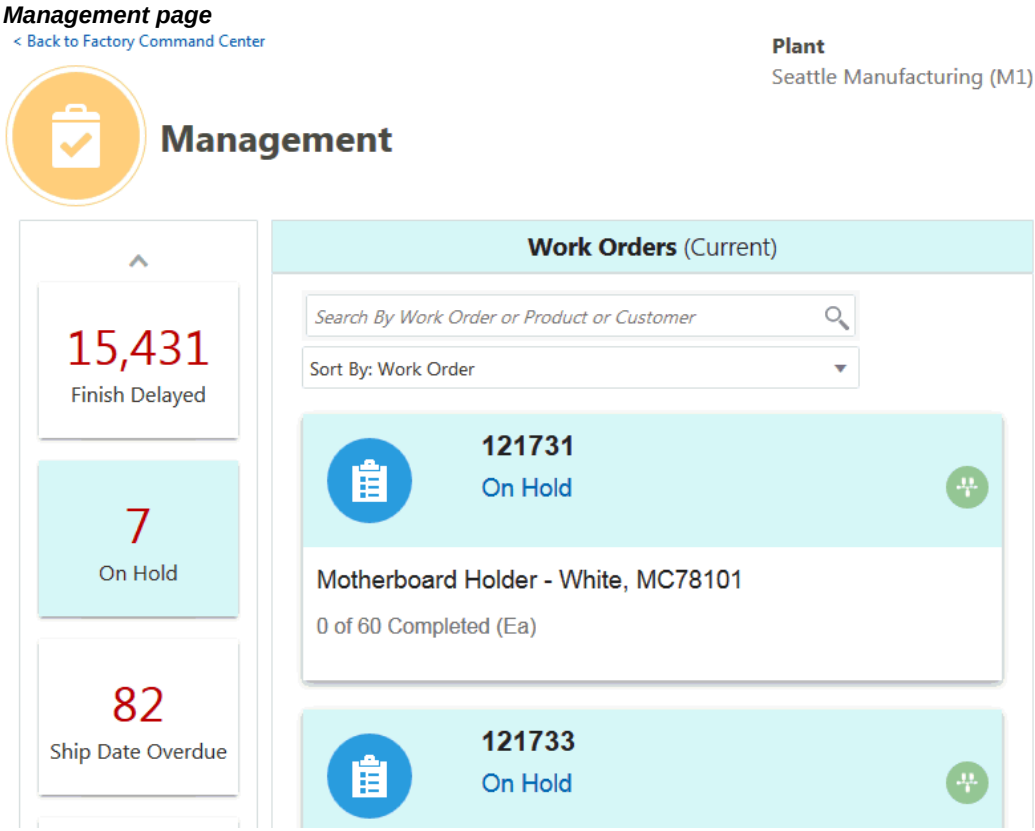
Filter information by work order, product, or customer using the Search By field.

The Management page displays the following details for a work order:

- Work order number.
- Work order status of In Progress (blue), On Hold, Start Delayed, or Finish Delayed (the last three statuses appear in red).
- Work progress status of On Track (blue), Start Delayed, Finish Delayed, or Operation Delayed (the last three statuses appear in red).
- Work order scheduled completion date and time, which appears in red if the work progress status is Finish Delayed.
- Item description and item number.
- Work Order Quantity, indicating the completed quantity out of the total start quantity.

Each work order tile on the Management page includes a Genealogy and Trace icon. Click this icon to view detailed lot and genealogy information about the work order in the Genealogy and Trace page.

*Management page*




## Viewing Material

The Factory Command Center displays the numerical values for the following KPIs listed under the Material category, which enable you to monitor items on the shopfloor.

- Shortage: Number of components/ingredients of any currently running open work orders for which the remaining quantity to be issued is greater than the on hand quantity available for production.
- Receipt Delayed: Number of components/ingredients with currently pending PO receipts that are past due beyond the Receipt Delay Tolerance (See Setting Up User Preferences, page 2-15 for more information).
- Expiring: Number of components/ingredients which are set to expire within the Expiration Time Fence (See Setting Up User Preferences, page 2-15 for more information).
- Production Returns: Number of components/ingredients returned from currently running open and recently completed work orders within the WIP Return Time Fence (See Setting Up User Preferences, page 2-15 for more information).

- Unallocated: Number of components/ingredients in any currently running open work orders for which the remaining quantity to be issued has not been allocated.

- Total: Number of components/ingredients related to the currently running work orders.

- With Prediction Alerts: Number of components/ingredients related to the currently running work orders that have prediction alerts where a component/ingredient parameter is a predictor.

Click a KPI to view details in the Material page. To view details of another KPI, click a KPI listed on the left side of the Material page.

> **Tip:** You can sort information for a KPI in the Material page using the following Sort By values:
>
> - Item Number
> - Item Description
> - Shortage
> - Expiring
>
> Filter information by item or item description using the Search By field.

The Material page displays the following details for components/ingredients:

- Item Number

- Item Description - appears in blue if the job status is In Progress or Released and in red if the job status is On Hold.

- Component/ingredient issue status of Fully Issued (green), Shortage (red), or Partially Issued (black).

- Work Orders - number of currently running open work orders that use the component/ingredient.

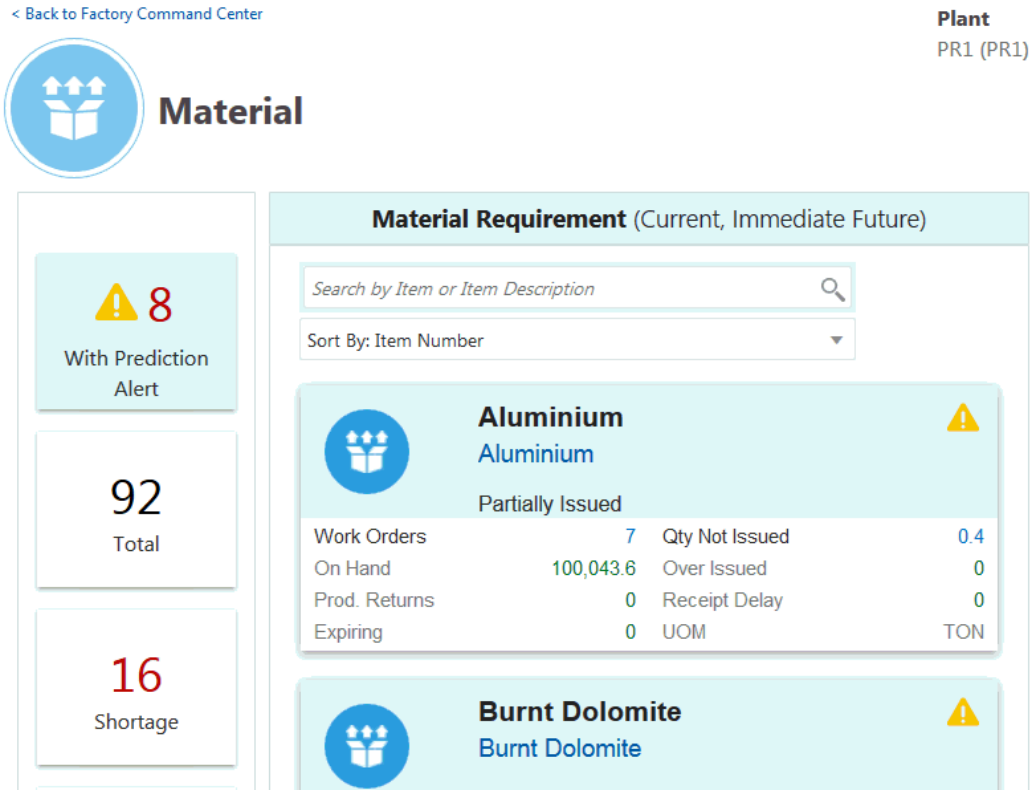
- Qty Not Issued - quantity of the specific component/ingredient not yet issued for the operations/steps requirements in all currently running open work orders.

- On Hand - quantity of the specific component/ingredient available in Inventory for production (including already allocated).

- Over Issued - excess quantity of the specific component/ingredient issued for the operations/steps requirements in all currently running open work orders.

- Prod. Returns - quantity of the specific component/ingredient currently issued and returned from currently running open and recently completed work orders within the WIP Return Time Fence.
- Receipt Delay - quantity of the specific component/ingredient with currently pending PO receipts that are past due beyond the Receipt Delay Tolerance.
- Expiring - quantity of the specific component/ingredient set to expire within the Expiration Time Fence.
- UOM - the primary unit of measure of the specific component/ingredient.

***Material page***




## Viewing Method

The Factory Command Center displays the numerical values for the following KPIs listed under the Method category, which enable you to monitor exceptions on the shopfloor.

- Material Exceptions.
- Resource Exceptions

- Processing Exceptions: The number of open exceptions of the type In Process/Assembly Exception.
- Quality Exceptions
- Other Exceptions
- Total: The number of all types of open exceptions.
- With Prediction Alerts: The number of exceptions related to the currently running work orders that have prediction alerts.

Click a KPI to view details in the Method page. To view details of another KPI, click a KPI listed on the left side of the Method page.

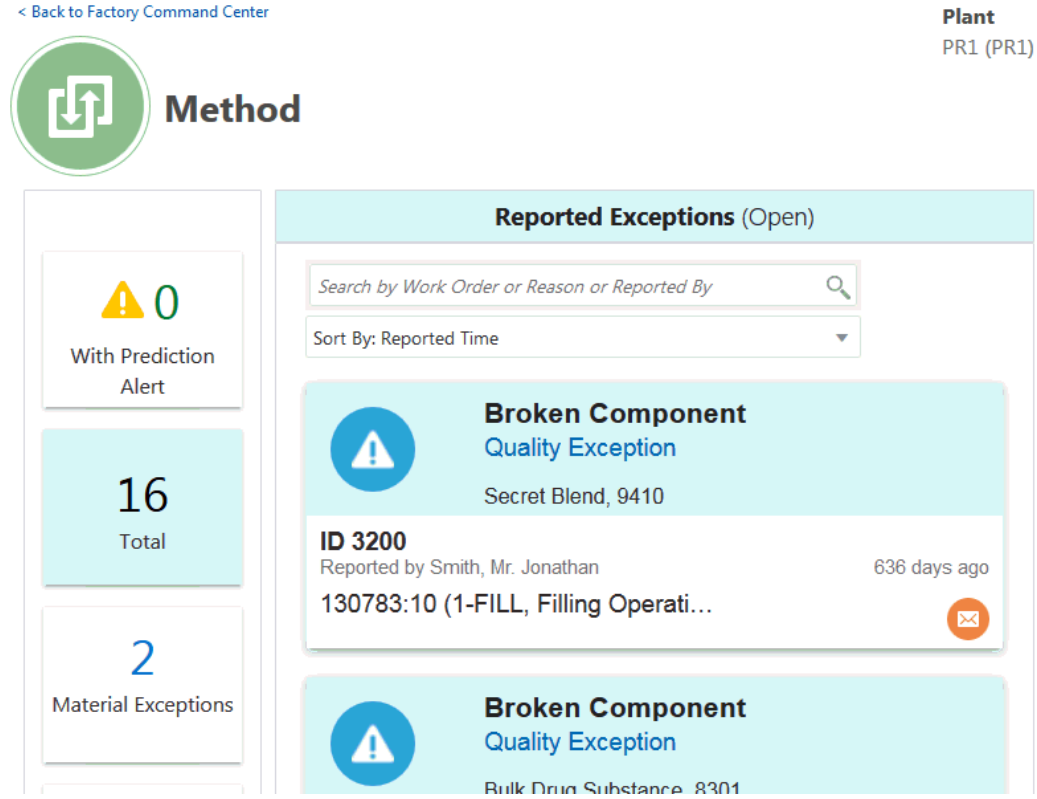
> **Tip:** You can filter information for a KPI in the Method page using the following Sort By values:
>
> - Reported Time
> - Item Number
> - Reported By
> - Work Order

The Method page displays the following details for open manufacturing operation/step execution related to currently running open work orders:

- Exception reason
- Exception type
- Exception details, such as:
  - Resource Exceptions: Resource Instance Name, Number of the Equipment or Operator Name in the case of an Operator Resource Exception.
  - Material Exception: Item or Component/Ingredient Description, and Number
  - Processing Exception: Assembly/Product/Recipe Description and the number of the context work order
  - Quality Exception: Assembly/Product or Component/Ingredient Description, and the number of the context work order
  - Other Exception: Assembly/Product/Recipe Description, and the number of the context work order

- Person who reported the exception
- Time elapsed since reporting the exception
- Exception Identifier/Number
- Context Work Order:operation seq/step(operation code, operation)
- Email icon, which opens the email application and defaults the email To field with the email ID of the person who reported the exception, the Subject field with the Exception ID, Exception Type, Exception Reason, and the email body with all the information given for the open exception.

***Method page***




**Note:** Support for KPIs for the 5 Ms in the Factory Command Center depends on the Business Entity Data ingestion method you are using to import the data.

The following table displays the KPI and the KPI tile attribute associated to each M of the Factory Command Center that are supported and not supported for process manufacturing organizations,

depending on the Business Entity Data ingestion method:

| Factory M | KPI | KPI Tile Attribute | Support Using CSV file Upload | Support Using REST Web Services Upload |
|---|---|---|---|---|
| Management | Ship Date Overdue | | No | No |
| | Pending Inspection | | No | No |
| | Quality Issues | | No | No |
| Material | Shortage | | No | No |
| | Receipt Delayed | | No | No |
| | Expiring | | No | No |
| | | Onhand | No | No |
| | | Component Summary Status | No | No |
| Method | | Exception Reported By | No | Yes |
| | Resource (Person) Exception | | No | Yes |
| | | Reason | No | Yes |
| Manpower | | | No | Yes |
| Machine | Under Maintenance | | No | No |

The following table displays the KPI and the KPI tile attribute associated to each M of the Factory Command Center that are supported and not supported for discrete manufacturing organizations, depending on the Business Entity Data ingestion method:

| Factory M | KPI | KPI Tile Attribute | Support Using CSV file Upload | Support Using REST Web Services Upload |
|---|---|---|---|---|
| Management | Ship Date Overdue | | No | No |
| | Pending Inspection | | No | No |
| Material | Shortage | | No | No |
| | Receipt Delayed | | No | No |
| | Expiring | | No | No |
| | | Onhand | No | No |
| | | Component Summary Status | No | No |
| Method | | Exception Reported By | No | Yes |
| | Resource (Person) Exception | | No | Yes |
| | | Reason | No | Yes |
| Manpower | Skills Mismatch | | No | No |
| | Not reported | | No | Yes |

| | | | |
|---|---|---|---|
| | Not Clocked in | No | Yes |
| Machine | Under Maintenance | No | No |

# A

# Navigation

## Oracle Adaptive Intelligent Apps for Manufacturing Navigation

The following table shows navigation paths to the functions and pages in Oracle Adaptive Intelligent Apps for Manufacturing from the Home page.

| Page Name | Navigation Path |
|---|---|
| Background Process | From the **User Name** drop-down list, select **Background Process**. |
| Business Entity Data | From the Home page, click **Insights** or **Predictions**, then **Data Ingestion**, then **Business Entity Data**. |
| Case Record Data | From the Home page, click **Insights** or **Predictions**, then**Data Ingestion**, then **Case Record Data**. |
| Change Your Password | From the **User Name** drop-down list, select **Profile**. In the **My Profile** console, click **Change My Password**. |
| Confusion Matrix | From the **Home** page, **Predictions** and then **Confusion Matrix**. |
| Create Dataset | From the Home page, click **Insights**, or **Predictions**, and then **Data Preparation**, and then **Create**. |
| Create Sensor Device Mappings | From the Sensor Devices page, click **Create**. |

| Page Name | Navigation Path |
|---|---|
| Create Time Series Feature Sets | From the Time Series Feature Sets page, click **Create**. |
| Data Preparation | From the Home page, click **Insights**, or **Predictions**, and then **Data Preparation**. |
| Dataset Information | From the Data Preparation page, click the actions link for a dataset, and then click **View Dataset Details**. |
| Define Key Performance Indicator | From the Key Performance Indicators page, click the **Plus** icon. |
| Factory Command Center | From the Home Page, click **Factory Command Center**. |
| Feature Significance | From the Insights page, click **Evaluation**, and then **Feature Significance**. |
| Genealogy and Trace | From the Home page, click **Genealogy & Trace**. |
| Insights | From the Home page, click **Insights**. |
| Key Performance Indicators | From the Home page, click **Insights**, or **Predictions**, then **Configuration**, and then **Key Performance Indicators**. |
| Machine | From the Factory Command Center page, click **Machine**. |
| Management | From the Factory Command Center page, click **Management**. |
| Manpower | From the Factory Command Center page, click **Manpower**. |
| Material | From the Factory Command Center page, click **Material**. |
| Method | From the Factory Command Center page, click **Method**. |

| Page Name | Navigation Path |
|---|---|
| Model | From the Home page, click **Insights**, then **Modeling**, and then **Create**.<br><br>From the Home page, click **Predictions**, then **Modeling**, and then **Create**. |
| Modeling | From the Insights page click **Modeling**.<br><br>From the Predictions page, click **Modeling**. |
| My Profile Details | From the **User Name** drop-down list, select **Profile**.<br><br>**Note:** This is an Oracle Identity Cloud Service page. See: Set Up or Modify Your Profile [https://docs.oracle.com/en/cloud/paas/identity-cloud/usids/set-or-modify-your-profile.html] |
| Network Viewer | From the Genealogy and Trace page, click the Search icon, then the **Network** option. |
| Organization Access | From the Setup page, click **Organization Access**. |
| Patterns & Correlations | From the Insights page, click **Evaluation**, and then **Patterns & Correlations**. |
| Predictive Analysis | From the Home page, click **Predictions**. |

| Page Name | Navigation Path |
|---|---|
| Predictor Detail | From the Quality Predictions page, click a Quality Predictions classification, then a work order, and then a predictor.<br><br>From the Serial Unit Quality Predictions page, click a Quality Predictions classification, then a serial unit, and then a predictor.<br><br>From the Yield Predictions page, click a Yield Predictions classification, then a work order, and then a predictor.<br><br>From the Serial Unit Yield Predictions page, click a Yield Predictions classification, then a work order, and then a predictor. |
| Quality Insight (detail page) | From the Quality Insights page, click the right arrow of one of the insights listed. |
| Quality Insights (summary page) | From the Insights page, in the Quality block, click the **More...** link. |
| Quality Predictions | From the Predictions page, click **Quality**. |
| Yield Insight (detail page) | From the Yield Insights page, click the right arrow of one of the insights listed. |
| Yield Insights (summary page) | From the Insights page, in the Yield block, click the **More...** link. |
| Sensor Devices | From the Home page, click **Insights** or **Predictions**, then **Configuration**, then **Sensor Devices**.<br><br>From the Home page, click **Insights** or **Predictions**, then **Data Ingestion**, then **Sensor Devices Data**.<br><br>From any page, click **Setup**, then **Sensor Devices**. |
| Sensor Time Series Features | From the Data Preparation page, click the actions link for a dataset, and then click**View Sensor Summary Results**. |

| Page Name | Navigation Path |
|---|---|
| Serial Unit Quality Insight (detail page) | From the Serial Unit Quality Insights page, click the right arrow of one of the insights listed. |
| Serial Unit Quality Insights (summary page) | From the Insights page, in the Serial Unit Quality block, click the **More...** link. |
| Serial Unit Yield Insight (detail page) | From the Serial Unit Yield Insights page, click the right arrow of one of the insights listed. |
| Serial Unit Yield Insights (summary page) | From the Insights page, in the Serial Unit Yield block, click the **More...** link. |
| Serial Unit Quality Predictions | From the Predictions page, click **Serial Unit Quality**. |
| Serial Unit Yield Predictions | From the Predictions page, click **Serial Unit Yield**. |
| Setup | From any page, click **Setup**. |
| Time Series Feature Sets | From the Home page, click **Insights**, or **Predictions**, then **Configuration**, and then **Time Series Feature Sets**. |
| Timeline Viewer | From the Genealogy and Trace page, click the Search icon, then the **Timeline** option. |
| Upload File | From the Sensor Devices page, click the **Data** tab, and then click **Upload**. |
| Update Sensor Device Mappings | From the Sensor Devices page, select a mapping, and then from the **Actions** link, click **Update**. |
| Update Time Series Feature Sets | From the Time Series Feature Sets page, select a times series features set, and then click **Update**. |
| User Preferences | From the **User Name** drop-down list, select **Preferences**. |

| Page Name | Navigation Path |
| --- | --- |
| Users & Role Assignments | From the Setup page, click **Users & Role Assignments**. |
| Yield Insight | From the Insights page, click **Yield**. |
| Yield Predictions | From the Predictions page, click **Yield**. |

# B

# Model Features

This appendix covers the following topics:

- Overview of Model Features
- Model Features for Process Manufacturing
- Model Features for Discrete Manufacturing

## Overview of Model Features

Select from the following features when creating a Feature Significance, Insights or Predictions model for a process manufacturing or discrete manufacturing plant. Depending on the type of model created, the model analysis reveals which features:

- influence the target measure results the most for Feature Significance models.
- enable the model to match historical work order data with a high level of factors influence for Insights models.
- best use historical work order data to predict outcomes using a Predictions model.

## Model Features for Process Manufacturing

Select from the following features when creating a model for a process manufacturing plant.

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Management/Operation | Operation Planned Duration | For each routing operation. | Numeric | Difference between planned completion date and planned start date of the step. |
| Management/Operation | Operation Actual Duration | For each routing operation. | Numeric | Difference between actual completion date and actual start date of the step. If the operation actual duration is not available while running the predictions for the batch, the operation planned duration is considered. |
| Management/Operation | Operation Duration Variance | For each routing operation. | Numeric | Difference between the planned step duration and the actual step duration. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Management/Operation | Operation Duration Variance % | For each routing operation. | Numeric | Ratio of the difference between the planned and actual duration to the planned duration of the step.<br><br>**Tip:** Best feature for analyzing the duration of non-uniform batch sizes. |
| Management/Operation | Operation Planned Offset | For each routing operation. | Numeric | Offset between the previous step's planned completion and the current step's planned start date. |
| Management/Operation | Operation Actual Offset | For each routing operation. | Numeric | Offset between the previous step's actual completion and the current step's actual start date. |
| Management/Operation | Operation Offset Variance | For each routing operation. | Numeric | Difference between the planned offset and the actual offset. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Management/Operation | Unplanned Operation Planned Offset | For each unplanned operation. | Numeric | Offset between the previous step's planned completion and the current (unplanned) step's planned start date. |
| Management/Operation | Unplanned Operation Actual Offset | For each unplanned operation. | Numeric | Offset between the previous step's actual completion and the current (unplanned) step's actual start date. |
| Management/Operation | Unplanned Operation Offset Variance | For each unplanned operation. | Numeric | Difference between the planned offset and the actual offset of the unplanned operation. |
| Management/Activity | Activity Plan Duration | For each routing operation activity. | Numeric | Difference between the planned completion date and the planned start date of the activity. |
| Management/Activity | Activity Actual Duration | For each routing operation activity. | Numeric | Difference between the actual completion date and the actual start date of the activity. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Management/Activity | Activity Duration Variance | For each routing operation activity. | Numeric | Difference between the planned activity duration and the actual activity duration. |
| Management/Activity | Activity Duration Variance % | For each routing operation activity. | Numeric | Ratio of Difference between planned and actual duration to the planned duration of the activity. |
| Method/Operation | Unplanned Operation Exist | For each unplanned operation. | Boolean | Determines whether any unplanned operation was added (a step not included in the routing definition). |
| Method/Operation | Unplanned Operation Actual Duration | For each unplanned operation. | Numeric | Difference between actual completion date and actual start date of the unplanned operation. |
| Method/Activity | Unplanned Activity Exist | For each unplanned activity. | Boolean | Determines whether any unplanned activity was added (activity not included in the routing definition). |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Method/Activity | Unplanned Activity Actual Duration | For each unplanned activity added to an existing operation. | Numeric | Difference between actual completion date and actual start date of the unplanned activity added to an existing operation. |
| Method/Exception | Material Exceptions Exist | For each routing operation. | Boolean (Y/N) | Determines whether there are any material exceptions created for the operation. |
| Method/Exception | Processing Exceptions Exist | For each routing operation. | Boolean (Y/N) | Determines whether there are any processing exceptions created for the operation. |
| Method/Exception | Resource Exceptions Exist | For each routing operation. | Boolean (Y/N) | Determines whether there are any resource exceptions created for the operation. |
| Method/Exception | Other Exceptions Exist | For each routing operation. | Boolean (Y/N) | Determines whether there are any other exceptions created for the operation. |
| Machine/Activity | Activity Primary Equipment | For each routing operation activity. | Categorical | Primary equipment of the activity. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Activity | Unplanned Equipment Exist | For each unplanned activity. | Boolean | Determines whether any unplanned equipment is in use (equipment not included in the routing definition). |
| Machine/Equipment | Equipment Actual Usage | For each routing operation activity equipment. | Numeric | Actual usage of the equipment. |
| Machine/Equipment | Unplanned Equipment Actual Usage | For each unplanned equipment instance of the existing operation/activity. | Numeric | Actual usage of the unplanned equipment. |
| Machine/Equipment | Unplanned Equipment Actual Quantity | For each unplanned equipment instance. | Numeric | Actual processing quantity for the unplanned equipment. |
| Machine/Equipment | Equipment Usage Variance | For each routing operation activity equipment. | Numeric | Difference between planned and actual usage of the equipment. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Equipment | Equipment Usage Variance % | For each routing operation activity equipment. | Numeric | The ratio of the difference between planned and actual equipment usage to the planned usage of the equipment.<br><br>**Tip:** Best feature for analyzing the usage variations between non-uniform batch sizes. |
| Machine/Equipment | Equipment Actual Quantity | For each routing operation activity equipment. | Numeric | Actual quantity processed by the equipment. |
| Machine/Equipment | Equipment Quantity Variance | For each routing operation activity equipment. | Numeric | Difference between planned and actual quantity processed by the equipment. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Equipment | Equipment Quantity Variance % | For each routing operation activity equipment. | Numeric | The ratio of the difference between planned and actual equipment quantity to the planned quantity of the equipment.<br><br>**Tip:** Best predictor for analyzing the quantity variations between non-uniform batch sizes. |
| Machine/Equipment | Equipment Instance Used | For each routing operation activity equipment instance. | Boolean (Y/N) | Determines whether the equipment instance is used or not. |
| Machine/Process Parameter | Parameter Value | For each routing operation activity equipment process parameter. | Numeric | Process parameter value of the equipment parameter. |
| Machine/Process Parameter | Parameter Value Deviation | For each routing operation activity equipment process parameter. | Numeric | Difference between the target and actual value of the process parameter. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Process Parameter | Parameter Value Outside Range | For each routing operation activity equipment process parameter. | Boolean | Determines whether the parameter actual value is outside the range specified in the routing process parameter definition. |
| Machine/Sensor Summary | <<Name of the Sensor Summary>><br><br>For example, the Average Context has Time Segment: Full, Parameter: Temperature, Equipment: Furnace, Activity: RUNTIME, Operation: MELT. | For each routing operation activity equipment with sensor parameters mapped in the device mapping and associated to the production analysis feature set. | Numeric | The computed summary function value.<br><br>For example, the average temperature of the full range of the Furnace temperature in the Runtime activity of the Melt operation. |
| Material/Quantity | Ingredient Actual Quantity | For each operation material line item. | Numeric | Actual quantity of the ingredient issued to the operation. |
| Material/Quantity | Ingredient Quantity Variance | For each operation material line item. | Numeric | Difference between planned and actual quantity of the ingredient issued to the operation. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Quantity | Ingredient Quantity Variance % | For each operation ingredient. | Numeric | Ratio of the difference between the planned and actual quantity to the planned quantity of the ingredient issued to the operation.<br><br>**Tip:** Best predictor for analyzing the material quantity variations between non-uniform batch sizes. |
| Material/Byproduct | Byproduct Quantity Variance | For each operation byproduct. | Numeric | Difference between the planned and actual quantity of the byproduct yielded to the operation. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Byproduct | Byproduct Quantity Variance % | For each operation byproduct. | Numeric | Ratio of the difference between the planned and actual quantity to the planned quantity of the byproduct yielded to the operation.<br><br>**Tip:** Best predictor for analyzing the byproduct quantity variations between non-uniform batch sizes. |
| Material/Lot Grade | Lot Grade Quantity | For each operation ingredient lot grade. | Numeric | Quantity of the ingredient lot of a specific grade issued to the operation. |
| Material/Quality Test | Ingredient Quality Result (Weighted Average) | For each operation ingredient supplier specification test. | Numeric | Weighted average of the test results of all the ingredient samples over each lot quantity issued. |
| Material/Quality Test | WIP Quality Result (Latest) | For each operation ingredient supplier specification test. | Numeric | Latest WIP quality sample test result. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Ingredient | Unplanned Ingredient Exists | For each unplanned ingredient. | Boolean | Determines whether an unplanned ingredient was added (an ingredient not included in the recipe definition). |
| Material/Ingredient | Unplanned Ingredient Actual Quantity | For each unplanned ingredient. | Boolean | Actual quantity of the unplanned ingredient issued to the operation. |
| Output | Work Order Yield | | | Ratio of the total output to the total input. |
| Output | Operation Yield | | | Ratio of the step output to the step input. |
| | <<Quality Test Name >> | | | Quality test value. |

## Model Features for Discrete Manufacturing

Select from the following features when creating a model for a discrete manufacturing plant.

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Management/Operation | Operation Actual Duration | For each routing operation. | Numeric | The actual duration for each operation that is part of an item's standard routing. This feature is not available for serial analysis. |
| Management/Operation | Operation Duration Variance % | For each routing operation. | Numeric | Ratio of the difference between the planned and actual duration to the planned duration for each operation included in the item's standard routing. This feature is not available for serial analysis. |
| Management/Operation Deviation | Unplanned Operations Exist | For each additional operation at work order level. | Categorical | Indicates whether the operation is an additional operation added during WIP and not part of an item's standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Management/Operation Deviation | Unplanned Operations Exist | At work order level. | Categorical | Indicates whether the operation is an additional operation added during WIP and not part of an item's standard routing. This feature is not available for serial analysis. |
| Management/Operation Deviation | Unplanned Operation Actual Duration | For each additional operation at work order level. | Numeric | The actual duration of an additional operation added during WIP and not part of an item's standard routing. This feature is not available for serial analysis. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Manpower/Operator | Operator Alternatives Used | For each routing operation. | Categorical | Indicates whether the operator resources used are primary resources, alternate resources defined in the operation as part of a standard routing, or unplanned resources used in the operation. The values can be all primary, all alternate, all unplanned, or combinations of primary, alternate, and unplanned. |
| Manpower/Operator | Operator Actual Usage | For each routing operation primary resource. | Numeric | The actual usage of an operator defined as a primary resource for an operation in the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Manpower/Operator | Operator Usage Variance % | For each routing operation primary resource. | Numeric | The ratio of the difference between planned and actual operator usage to the planned usage of the operator. The operator is defined as a primary resource for an operation in the standard routing. |
| Manpower/Operator | Operator Instance Worked | For each routing operation primary resource. | Categorical | Indicates the actual operator instance defined as a primary resource or alternate resource for an operation in the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Manpower/Operator Operation Deviation | Operator Alternatives Used (Unplanned Operation) | For each additional operation at work order level. | Categorical | Indicates whether operator resources used are primary resources, alternate resources defined in the operation as part of the standard routing, or unplanned resources used in the operation.<br><br>This feature displays values only if additional operations are added in WIP and are not part of the standard routing. |
| Manpower/Operator Operation Deviation | Operator Actual Usage (Unplanned Operation) | For each additional operation actual resource. | Numeric | The actual usage of an operator in the operation when the operation is added as an additional operation during WIP and is not part of the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Manpower/Operator Deviation | Unplanned Operator Actual Usage | For each routing operation additional resource. | Numeric | The actual usage of an operator in the additional operations added during WIP and not included in the standard routing. |
| Manpower/Operator Alternate | Alternate Operator Actual Usage | For each routing operation alternate resource. | Numeric | The actual usage of an operator defined as an alternate resource for an operation in the standard routing. |
| Manpower/Operator Alternate | Alternate Operator Usage Variance % | For each routing operation alternate resource. | Numeric | The ratio of the difference between planned and actual alternate operator usage to the planned usage of the alternate operator. The operator is defined as an alternate resource for an operation in the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Equipment | Equipment Alternatives Used | For each routing operation. | Categorical | Indicates whether the equipment resources used are primary resources, alternate resources defined in the operation as part of standard routing, or unplanned resources used in the operation. The values can be all primary, all alternate, all unplanned, or combinations of primary, alternate, and unplanned. |
| Machine/Equipment | Equipment Actual Usage | For each routing operation primary resource. | Numeric | The actual usage of equipment in the operation defined as a primary resource in the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Equipment | Equipment Usage Variance % | For each routing operation primary resource. | Numeric | The ratio of the difference between planned and actual equipment usage to the planned usage of the equipment. The equipment is defined as a primary resource for an operation in the standard routing. |
| Machine/Equipment | Equipment Instance Used | For each routing operation primary resource. | Categorical | Indicates the actual equipment instance defined as the primary resource or alternate resource for an operation in the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Equipment Operation Deviation | Equipment Alternatives Used (Unplanned Operation) | For each additional operation at work order level. | Categorical | Indicates whether operator resources used are primary resources, alternate resources defined in the operation as part of the standard routing, or unplanned resources used in the operation. The values can be all primary, all alternate, all unplanned, or combinations of primary, alternate, and unplanned.<br><br>This feature displays values only if additional operations are added in WIP and are not part of the standard routing. |
| Machine/Equipment Operation Deviation | Equipment Actual Usage (Unplanned Operation) | For each additional operation actual resource. | Numeric | The actual usage of equipment in the operation when the operation is added as an additional operation during WIP and is not part of the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Equipment Deviation | Unplanned Equipment Actual Usage | For each routing operation additional resource. | Numeric | The actual usage of equipment in the additional operations added during WIP and not included in the standard routing. |
| Machine/Equipment Alternate | Alternate Equipment Actual Usage | For each routing operation alternate resource. | Numeric | The actual usage of equipment defined as an alternate resource for an operation in the standard routing. |
| Machine/Equipment Alternate | Alternate Equipment Usage Variance % | For each routing operation alternate resource. | Numeric | The ratio of the difference between planned and actual alternate equipment usage to the planned usage of the alternate equipment. The equipment is defined as an alternate resource for an operation in the standard routing. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Machine/Sensor Summary | <<Name of the Sensor Summary>><br><br>For example, Maximum Context includes Time Segment: Segment (0-30 mins), Parameter: Vibration, Equipment: Robot, Operation: Assembly. | For each routing operation equipment with sensor parameters mapped in the device mapping and associated to the production analysis feature set. | Numeric | Feature extracted from time series data by dividing the time series data into segments and applying functions based on the time series feature set's definition. For example, the maximum vibration of the robot equipment in the assembly operation during the (0-30 mins) segment. |
| Method/Exception | Operation Exceptions Exist (Unplanned Operation) | For each additional operation at work order level. | Categorical | Indicates whether there are any exceptions in the additional operations added in WIP which are not part of the standard routing.<br><br>Acceptable values are Yes or No. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Method/Operation Exception Exist | Operation Exceptions Exist | For each routing operation. | Categorical | Indicates whether there are any exceptions in the operations that are part of the standard routing. Acceptable values are Yes or No. |
| Material/Quantity | Component Alternatives Used | For each routing operation. | Categorical | Indicates whether the components used are primary components, substitute components defined in standard bills of material, or unplanned components. Acceptable values are all primary, all substitute, all unplanned, and combinations of primary, substitute, and unplanned. |
| Material/Quantity | Component Actual Quantity | For each component. | Numeric | The actual quantity of a primary component issued to a WIP job. This feature is limited to Push type components only. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Quantity | Component Quantity Variance % | For each component. | Numeric | The variance percentage of a primary component quantity issued to a WIP job. This feature is limited to Push type components only. |
| Material/Quantity | Highest Variance Component | For each routing operation. | Categorical | Provides the name of the component with the maximum usage quantity deviation in a WIP job. This feature is limited to Push type components only. |
| Material/Lot Number | Count of Lots Consumed | For each routing operation lot controlled component. | Numeric | Provides the count of component lots that are issued to the WIP job. |
| Material/Supplier | Supplier of Lot | For each routing operation lot controlled buy component. | Categorical | Provides the name of the supplier that provides the maximum number of components issued in a lot for the WIP job. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Supplier | Multiple Suppliers for Lot | For each routing operation lot controlled buy component. | Categorical | Indicates if there are multiple suppliers who supplied the component lots that are issued to a WIP job. Acceptable values are Yes or No. |
| Material/Substitute Component | Substitute Component Actual Quantity | For each substitute component at routing level. | Numeric | The actual quantity of substitute component issued to a WIP job. This feature is limited to Push type components only. |
| Material/Substitute Component | Substitute Component Quantity Variance % | For each substitute component. | Numeric | The variance percentage of a substitute component quantity issued to a WIP job. This feature is limited to Push type components only. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Component Deviation | Unplanned Component Actual Quantity | For each additional component. | Numeric | The actual quantity of an unplanned component that is issued to a WIP job. Unplanned components are not a part of standard bills of material. This feature is limited to Push type components only. |
| Material/Component Operation Deviation | Component Actual Quantity (Unplanned Operation) | For each additional operation component. | Numeric | The actual quantity of a primary component issued to a WIP job for an additional operation added in WIP and not part of the standard routing. This feature is limited to Push type components only. |
| Material/Quality Test | WIP Quality Result (Latest) | For each operation component supplier specification test. | Numeric | WIP quality sample test result average. |

| Category/Subcategory | Feature | Level | Numeric/Categorical | Description |
|---|---|---|---|---|
| Material/Quality Test | Component Quality Result (Weighted Average) | For each operation component supplier specification test. | Numeric | The weighted average of test results for component samples of each lot quantity issued. |
| Output | Operation First Pass Yield | For each operation. | Numeric | The percentage of good quantity completed compared to the total quantity completed in the work order operation. |
| Output | Work Order First Pass Yield | For each work order. | Numeric | The percentage of good quantity completed compared to the total quantity of the work order. |
| Output | Serial Unit Operation Yield | For each operation for a serial unit. | Categorical | The operation yield of a serial unit. Acceptable values are Pass or Fail. |
| Output | Serial Unit Yield | For each serial unit. | Categorical | It The yield of a serial unit. Acceptable values are Pass or Fail. |
| Output | <Quality Element> | - | - | Quality test value. |

# C

# Algorithms

This appendix covers the following topics:

- Overview of Algorithms
- Feature Significance Model Algorithms
- Insight Model Algorithms
- Predictions Model Algorithms

## Overview of Algorithms

Select an algorithm to use when creating a model. The algorithm selected uses the model features selected to predict a specific target measure. The output from each model varies depending on the algorithm and model features selected. The following algorithm descriptions provide a general overview of each algorithm choice for each model type.

## Feature Significance Model Algorithms

When creating a Feature Significance model, you can choose from either of the following algorithms:

- Random Forest Classifier - Random Forest is a powerful machine learning algorithm. The Random Forest algorithm builds a number of decision tree models and predicts using the ensemble of trees.

- Chi Square - A chi-square test for independence compares two variables in a table (in our case, one of the selected features and the quality or yield result) to see if they are related. In a more general sense, it tests to see whether distributions of each variable differ from each other. The algorithm compares multiple feature/result variable pairs at once to determine which features have a relationship with the chosen result.

## Random Forest Algorithm

For a detailed explanation of the Random Forest algorithm, see "Random Forest [https://docs.oracle.com/en/database/oracle/oracle-database/18/dmcon/random-forest.html#GUID-B6506C33-8555-4181-993F-CD7D48B4DA3C]" in *Oracle Data Mining Concepts* and visit the Apache Spark website at Random Forests [http://spark.apache.org/docs/latest/mllib-ensembles.html#random-forests].

You can set the following parameters for a random forest algorithm:

- Maximum Depth - Maximum depth of each tree.
- Minimum No. of Trees - The minimum number of trees to build before calculating the maximum voting or averages of predictions. A higher number of trees produces more accurate results, but requires more processing time. Choose as high a value as possible considering your processor's capabilities, because a high value results in stronger and more stable predictions.
- Minimum Importance - Specify a minimum importance value to filter for features with greater than the minimum level of importance.

For Random Forest algorithm examples, visit the Apache Spark website at Random Forests [http://spark.apache.org/docs/latest/mllib-ensembles.html#random-forests].

## Chi-Square Algorithm

The Chi-Square test is used to test the independence of two events. More specifically, to determine feature significance, the Chi-Square test is used to test whether the occurrence of a specific term (input feature) and the occurrence of a specific class (output variable) are independent. The algorithm generates a p-value which determines how likely or unlikely it is to have a NULL hypothesis (where the input and output are completely independent). The lower the p-value, the more unlikely the NULL hypothesis, indicating a relationship between the input and the output variables. Higher p-values indicate the likelihood of a NULL hypothesis (no relationship). You must specify the acceptable minimum importance for the model. Features with an importance above the set minimum limit are not considered significant to the quality or yield result. The system ranks all features with a p-value below this threshold, with the lowest p-value ranked the highest.

Features are ranked as follows:

| Minimum Importance (p-value) | Strength of Relationship |
|---|---|
| 0 to 0.01 | Very Strong |

| Minimum Importance (p-value) | Strength of Relationship |
|---|---|
| 0.01 to 0.05 | Strong |
| 0.05 to Minimum Importance (default value = 0.1) | Weak |
| > Minimum Importance | No relationship |

For more details about the Chi-Square algorithm, visit the Apache Spark website at Hypothesis testing [https://spark.apache.org/docs/1.6.3/mllib-statistics.html#hypothesis-testing].

## Insight Model Algorithms

When creating an Insight model, you can choose either of the following algorithms:

- Apriori - performs market basket analysis by identifying co-occurring items (frequent itemsets) within a set. Apriori finds rules with support greater than a specified minimum support and confidence greater than a specified minimum confidence. .

- Decision Tree - extracts predictive information in the form of human-understandable rules. The rules are if-then-else expressions; they explain the decisions that lead to the prediction. Predictions models also use the Decision Tree algorithm and Feature Significance models use a variation of the Decision Tree algorithm, called Random Forest.

### Apriori Algorithm

Association rules set the minimum level of predictability that is acceptable for this algorithm and data set. For a detailed explanation of the Apriori algorithm, see "Apriori [https://docs.oracle.com/en/database/oracle/oracle-database/18/dmcon/apriori.html#GUID-B7D12599-FB4C-45E3-BCE4-E54A3C6F0E64]" in *Oracle Data Mining Concepts*. Enter acceptable values for the following three parameters:

- Maximum Rule Length - Defines the maximum number of features/predictors that can influence the model's output (quality or yield). The default value is 4 and the highest allowed value is 20.

- Minimum Confidence - Specifies the minimum conditional probability of a target outcome, given that you have certain other outcomes in your data set. The default value is 0.75.

- Minimum Support - Defines the required minimum percentage of the data containing the target outcome. The default value is 0.2.

### Decision Tree Algorithm

For a detailed explanation of the Decision Tree algorithm, see "Decision Tree [https://docs.oracle.com/en/database/oracle/oracle-database/18/dmcon/decision-tree.html#GUID-14DE1A88-220F-44F0-9AC8-77CA844D4A63]" in *Oracle Data Mining Concepts*.

You can set the following parameters for a decision tree algorithm:

- Minimum Records to Node - the minimum number of records that must be present in a node for a split to occur. The default value is 10.
- Minimum Records to Split - when the node splits the data into branches, this is the minimum number of records for each branch. The default value is 20.
- Maximum Depth - the number of branch levels. The default value is 7.
- Impurity Metric - Use either entropy or Gini. Gini is the default value. Purity metrics, also known as homogeneity metrics, asses the quality of alternative split conditions and select the one that results in the most homogeneous child nodes. Purity refers to the degree to which the resulting child nodes are made up of cases with the same target value. The objective is to maximize the purity in the child nodes.

## Predictions Model Algorithms

When creating a Predictions model, you can choose from either of the following algorithms:

- Decision tree - Decision trees extract predictive information in the form of human-understandable rules. The rules are if-then-else expressions; they explain the decisions that lead to the prediction. Insight models also use the Decision Tree algorithm and Feature Significance models use a variation of the Decision Tree algorithm, called Random Forest.
- SVM (Support Vector Machine) - Distinct versions of Support Vector Machines (SVM) use different kernel functions to handle different types of data sets. Linear and Gaussian (nonlinear) kernels are supported. SVM classification attempts to separate the target classes with the widest possible margin. SVM regression tries to find a continuous function such that the maximum number of data points lie within an epsilon-wide tube around it.

## Decision Tree Algorithm

For a detailed explanation of the Decision Tree algorithm, see "Decision Tree [https://docs.oracle.com/en/database/oracle/oracle-database/18/dmcon/decision-tree.html#GUID-14DE1A88-220F-44F0-9AC8-77CA844D4A63]" in *Oracle Data Mining Concepts*.

You can set the following parameters for a decision tree algorithm:

- Minimum Records to Node - the minimum number of records that must be present in a node for a split to occur. The default value is 10.
- Minimum Records to Split - when the node splits the data into branches, this is the minimum number of records for each branch. The default value is 20.
- Maximum Depth - the number of branch levels. The default value is 7.
- Impurity Metric - Use either entropy or Gini. Gini is the default value. Purity metrics, also known as homogeneity metrics, asses the quality of alternative split conditions and select the one that results in the most homogeneous child nodes. Purity refers to the degree to which the resulting child nodes are made up of cases with the same target value. The objective is to maximize the purity in the child nodes.

## SVM Algorithm

For a detailed explanation of the SVM algorithm, see "Support Vector Machines [https://docs.oracle.com/en/database/oracle/oracle-database/18/dmcon/support-vector-machines.html#GUID-FD5DF1FB-AAAA-4D4E-84A2-8F645F87C344]" in *Oracle Data Mining Concepts*.

Specify the following SVM parameters:

- Active Learning - Value defaults to TRUE. This restricts the SVM algorithm to use the most informative samples of the data rather than attempting to use the whole data set.
- Convergence Tolerance - Value defaults to 0.0001. Setting a larger tolerance value for model convergence criteria enables faster model building, but it is less accurate. This optimization parameter is used to minimize the prediction loss of a target value while building a model.

# D

# REST Web Services

This appendix covers the following topics:

- Overview of Oracle Adaptive Intelligent Apps for Manufacturing REST Web Services
- Registering the AIAMFG Client Application in Oracle Identity Cloud Service
- Accessing External REST Web Services Using Postman
- Using Inbound REST Web Services
- Using Outbound REST Web Services

## Overview of Oracle Adaptive Intelligent Apps for Manufacturing REST Web Services

Representational State Transfer, or REST web services, provide interoperability between computer systems and the web. REST web services enable requesting systems to access and manipulate resources. There are two general categories of REST web services used in Adaptive Intelligent Applications for Manufacturing, inbound and outbound. Inbound web services enable a user to import data into Adaptive Intelligent Applications for Manufacturing tables from third party systems using the POST or PUT methods or to alter data using the DELETE method. Outbound web services enable a user to export Adaptive Intelligent Applications for Manufacturing data to third party systems using the GET method. Within the inbound and outbound categories, REST web services are further categorized into the sets of web services listed below.

#### Inbound Web Service Sets

- Factory Command Center Method exceptions
- Factory Command Center Manpower operators (person-type resources)
- Custom KPIs for Factory Command Center
- Custom Predictors for Insights and Predictions analysis

- Custom events for Timeline Viewer
- Custom nodes for Network Viewer

**Outbound Web Service Sets**

- Insights
- Predictions

The following tables show the web services included in each set and their available actions:

***Inbound Web Service Sets***

| Set of Web Services | Web Service Name | Description |
|---|---|---|
| Factory Command Center Method exceptions | Create/Update Exception (wipexception) | Use to create an exception, update an exception reason, or to update an exception as resolved.<br><br>Only exceptions from Oracle Manufacturing Execution System for Discrete Manufacturing are automatically synchronized to the Factory Command Center Method page. Manufacturers using any other manufacturing execution system (MES), including Oracle Manufacturing Execution System for Process Manufacturing, must use a REST web service to import or synchronize their exceptions into the Method page. |

| Set of Web Services | Web Service Name | Description |
|---|---|---|
| Factory Command Center Manpower operators (person-type resources) | Create Resource (createresource) | Creates a resource.<br><br>Person-type resources defined in Oracle E-Business Suite (EBS) products used by discrete manufacturers are automatically synchronized to the Factory Command Center Manpower page, but discrete and process manufacturers (even if using process manufacturing EBS products) using any other system to track manpower must use a REST web service to import or synchronize person-type resources into the Manpower page. |
| | Update Resource (changeresource) | Updates a resource. |
| | Find Resource (getresource) | Finds a resource. |
| | Create Resource-Employee Association (createresempassoc) | Associates a person to a resource. |
| | Update Resource-Employee Association (changeresempassoc) | Updates a person resource association. |
| | Find Resource-Employee Association (getinstanceId) | Finds a person resource association. |
| | Remove Resource-Employee Association (removeresempassoc) | Removes the association between a person and a resource. |
| | Employee Job assignment (createempjobassignment) | Assigns a job to an employee. |

| Set of Web Services | Web Service Name | Description |
|---|---|---|
| | Change/Update Employee Job Assignment (changeempjobassignment) | Updates a job assignment. |
| | Find Jobs Assigned to Employee (getempassignmentforjob) | Find jobs assigned to an employee. |
| | Find Employees Assigned to a Job (getjobassignmentforemp) | Find employees assigned to a job. |
| | Resource Equipment association (createresequpassoc) | Associates an equipment instance to a resource. |
| | Update Resource Equipment association (updatersrcequipment) | Updates an equipment instance to resource association. |
| | Find Resource Equipment association (getreseqpassoc) | Finds an equipment instance and resource association. |
| | Remove Resource Equipment association (removeresequipassoc) | Removes the association between an equipment instance and a resource. |
| | Log Resource In Time (timelogresourceintime) | Inserts a shift-in/clock-in record for an employee. |
| | Log Resource Out Time (logresourceouttime) | Inserts a shift-out/clock-out record for an employee. |
| | Find Employee Shift-In Details (isresshiftedin) | Find employee shift details. |
| Custom KPIs for Factory Command Center | Define KPI | Create a KPI. |
| | defineKPIValue | Add a KPI value to a KPI. |
| | Update KPI | Update a KPI. |

| Set of Web Services | Web Service Name | Description |
|---|---|---|
| | Delete KPI | Delete a KPI. |
| | Fetch KPI List | Fetch all KPI parameters. |
| | fetchKpisByCategory | Fetch KPI's by category. Use the last parameter as a search parameter where you can pass a search string. |
| | fetchKpisByParamName | Fetch KPI's by parameter name. Use the last parameter as a search parameter where you can pass a search string. |
| Custom Predictors for Insights and Predictions analysis | Predictor Get Rest API | Use this API to retrieve:<br>• a list of all custom predictors.<br>• a list of all custom predictors for a given organization.<br>• details about a custom predictor.<br>• details about a custom predictor based on the predictor's ID and organization. |
| | Predictor Insert Rest API | Create custom predictor data. |
| | Predictor Update Rest API | Update specific custom predictor details for a given custom predictor. |
| | Predictor Delete Rest API | Delete existing custom predictor data. |

| Set of Web Services | Web Service Name | Description |
|---|---|---|
| | Data Get Rest API | Either retrieves data for all predictors or retrieves data for specific predictor, based on Case ID or Job ID. |
| | Predictor Data Insert Rest API | Creates predictor data details. |
| | Predictor Data Update Rest API | Updates data details for a specific predictor. |
| | Predictor Data Delete Rest API | Deletes specific data details for a specific predictor based on Case ID. |
| Custom events for Timeline Viewer | Equipment Timeline REST API | Creates additional events for the Equipment Timeline. |
| | Serial Timeline REST API | Creates additional events for the Serial Timeline. |
| | Work Order Timeline REST API | Creates additional events for the Work Order Timeline. |
| | Lot Timeline REST API | Creates additional events for the Lot Timeline |
| Custom entities for the Timeline Viewer | User Defined Entity REST API | Creates custom entities for use in the Timeline Viewer. |
| | User Defined Entity Instance REST API | Creates custom entity objects for use in the Timeline Viewer. |
| Custom nodes for Network Viewer | Create Network | Creates nodes, links between the provided nodes, and/or user defined indicator icons. |

***Outbound Web Service Sets***

| Set of Web Services | Web Service Name | Description |
|---|---|---|
| Insights | Get Quality Insights | Provides quality insights by organization. |
| | Get Quality Insight Search Form | Provides syntax to use with the Get Quality Insights web service. |
| | Get Yield Insights | Provides yield insights by organization. |
| | Get Yield Insight Search Form | Provides syntax to use with the Get Yield Insights web service. |
| Predictions | Get Predictions | Provides predictions by organization. |
| | Get Prediction Detail | Provides details for a particular prediction. |

Before a user can access the REST web services used in Adaptive Intelligent Applications for Manufacturing, a user with an Oracle Identity Cloud Service Identity Domain Administrator or Application Administrator role must register the AIAMFG client application. Users can then use the AIAMFG client application to access the REST web services.

## Registering the AIAMFG Client Application in Oracle Identity Cloud Service

Registering the AIAMFG client application in Oracle Identity Cloud Service enables a user to obtain the credentials (Client ID and Client Secret) used for authentication in REST web service calls. The steps to register the AIAMFG client application generally are similar to the steps provided in the online tutorial Using the Oracle Identity Cloud Service REST APIs with Postman [https://www.oracle.com/webfolder/technetwork/tutorials/obe/cloud/idcs/idcs_rest_postman_obe/rest_postman.html], but since there are differences, the specific steps are listed below.

**Prerequisites**: Verify that you meet the expectations listed under What Do You Need? [https://www.oracle.com/webfolder/technetwork/tutorials/obe/cloud/idcs/idcs_rest_postman_obe/rest_postman.html#WhatDoYouNeed?] before you begin.

1. In the Oracle Identity Cloud Service administration console, expand the **Navigation Drawer**, click **Applications**, and then click **Add**.

2. In the **Add Application** dialog box, select Confidential Application.

3. In the **App Details** section on the **Add Trusted Application** page, enter an application name and description, and then click **Next**.

   - **Application Name**: AIAMfg_ClientApp
   - **Description**: Client App to access REST APIs provided by AIAMfg Application whose url starts with /aimfgapi.

4. Select **Configure this application as a client now**, and then, in the **Authorization** section that appears, select only Client Credentials as the **Allowed Grant Type**.

5. Click **Add Scope**.

6. On the Select Scope page, select the **AIAppsforManufacturing** resource, then the right arrow icon. This displays the available scopes for the resource.

7. Select the scope that ends with **/aimfgapi**, then click **Add**.

   **Important:** Copy the complete URL (scope) that ends with /aimfgapi to a text file. Later, provide this scope for use as the access token and base URL for the REST APIs.

8. Click **Next** on the page that shows AIAppsforManufacturing as a resource.

9. In the Resources page, **Skip for later** is selected by default. Click **Next**.

10. 11. In the Web Tier Policy page, **Skip for later** is selected by default. Click **Next**.

11. Click **Next** until you reach the **Authorization** page. Click **Finish**.

12. In the Authorization page, do not make any changes. Click **Finish**.

13. The Application Added page appears, displaying the **Client ID** and **Client Secret**. Copy both of these field values to a text file. Save and store the text file securely.

14. Click **Activate**. In the pop-up window, click **Activate Application**.

    The client application is now ready to use.

## Accessing External REST Web Services Using Postman

After registering the client application, use a third-party tool, such as Postman, to access

the AIAMFG REST web services.

1. Open Postman.

2. Create a new request.

   Click **New**, then **Request**.

3. In the Save Request page, enter values in the following fields, then click **Save to (collection or folder name)**.

   - Select the method of importing or exporting data (GET, POST, PUT, or DELETE). This example uses POST.

   - **Request Name**: AIAMFG REST Web Services

   - Search for and select a collection or folder. If needed, click **Create Collection**, then select the collection you created.

4. Select the request, then the **Authorization** tab.

   In the **Authorization** tab, select OAuth 2.0 in the **Type** field.

5. Click **Get New Access Token**.

6. In the Get New Access Token page, enter the following field values:

   - **Token Name**: Enter any name.

   - **Grant Type:** Client Credentials

   - **Access Token URL**: An Oracle Identity Cloud Service token URL, in the format <IDCS host>/oauth2/v1/token. For example, https://idcs-xyxyxy.hostname.com/oauth2/v1/token.

   - In the following three fields, paste the value copied when Registering the AIAMFG Client Application in Oracle Identity Cloud Service, page D-7.

     - Client ID

     - Client Secret

     - Scope

   - Client Authentication: Send Client Credentials in body

7. Click **Request Token**.

   A pop-up window opens with the Access Token, which is a long string. Copy the long string to a text file. Close the pop-up window.

**Warning:** The token is only valid for 1 hour. After that, you must request another token.

8. In the new request window, click the Headers tab. Enter the following key values:

| Key | Value |
|---|---|
| Content-Type | application/json |
| Authorization | Bearer <the access token copied from the previous step>. For example: Bearer eyJ4NXQjUzI1Nil6lII... |
| language | D, E, ESA, F, FRC, I, JA, KO, NL, PTB, RU, TR, or US. If no value is provided, the language used is US. |

9. In the Post field, enter the REST API URL.

   To derive the REST API URL, use the scope created when Registering the AIAMFG Client Application in Oracle Identity Cloud Service, page D-7 and a REST web services URL. For example:

   - Scope: https://...../aimfgapi
   - REST web services URL: <host:>/aimfgapi/vof/v1/fsm/wipexception

   Based on the above values, the REST API URL is https://..... /aimfgapi/vof/v1/fsm/wipexception.

10. Click the Body tab. Select the following field values:
    - Input data type: raw
    - Format: JSON

11. Click **Send**.

# Using Inbound REST Web Services

## Create/Update Exception (wipexception)

URL: <host:>/aimfgapi/vof/v1/fsm/wipexception

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| ExceptionId | Resource exception identifier. | String | A unique value is required. |
| Action | Use CREATE for creating an exception, UPDATE for updating the exception reason, and RESOLVE for updating the exception as resolved. | String | A value of CREATE, UPDATE or RESOLVE is required. |
| OrganizationId | Organization ID | String | A value is required for either organizationCode or OrganizationId. |
| OrganizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| WorkOrderNumber | WIP entity name or batch number. | String | Yes |
| OperationNumber | Operation number. | String | Yes |
| ExcepType | Exception types are defined as part of the FND Lookup type WIP_EXCEPTION_TYPE. | String | A value is required for either ExcepType or ExcepTypeCode. |
| ExcepTypeCode | Exception types are defined as part of the FND Lookup type WIP_EXCEPTION_TYPE. | String | A value is required for either ExcepType or ExcepTypeCode. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| RsrcSeqNo | Details of the resource related to the resource exception. | String | Required if the Resource Type is Resource Exceptions. |
| RsrcId | Details of the resource related to the resource exception. | String | Required if the Resource Type is Resource Exceptions. |
| ItemId | Item details related to material exception. | String | Required if the Resource Type is Component Exceptions. |
| Reason | Description of the reason. | String | A value is required for either Reason or ReasonCode. |
| ReasonCode | Reason codes defined as part of the FND Lookup type WIP_EXCEPTION_REASON_CODES. | String | A value is required for either Reason or ReasonCode. |
| ReportedBy | User ID of the reported user. | String | No |
| Note | Note for the exception. | String | No |

**ExamplePayload**

```
Input:
  {
  "ExceptionId" : "9992",
      "Action" : "CREATE",
      "OrganizationId": "207",
     "WorkOrderNumber" : "61974",
       "OperationNumber" : "20",
     "ItemId" : "151",
     "ExcepType" : "Component",
     "Reason" : "Broken Component",
     "ReportedBy" : "57"
   }
Output:
{
    "action": "CREATE",
    "excepType": "Component",
    "exceptionId": 9992,
    "itemId": 151,
    "operationNumber": 20,
    "organizationId": 207,
    "reason": "Broken Component",
    "reportedBy": 57,
    "returnExceptionID": 9992,
    "returnStatus": "Success",
    "workOrderNumber": "61974"
}
```

## Create Resource (createresource)

URL: <host:>/aimfgapi/vof/v1/fsm/setup/common/createresource

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| OrganizationId | Organization ID. | String | A value is required for either organizationCode or OrganizationId. |
| resourceCode | Resource code. | String | Yes |
| resourceType | Resource Type. Set to 2 for Manpower. | String | Yes |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| description | Description of the resource. | String | No |
| disableDate | Resource disable date. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ" | Date | No |

**ExamplePayload**

```
Input:
{
   "resourceCode" : "LBR-MOLD28",
    "organizationCode" : "PR1",
    "description" : "Molding Labor",
    "disableDate" : "2017-12-12T14:00:00.000Z",
     "resourceType":"2"
}

Output:
{
    "resourceId": -1081,
    "status": "success"
}
```

## Update Resource (changeresource)

URL:<host:>/aimfgapi/vof/v1/fsm/setup/common/changeresource

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| newResourceCode | Enter a new resource code if you want to update the existing resource code. | String | No |
| disableDate | The date the resource is disabled. Enter a value if you want to update it. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ" | Date | No |
| description | Description of the resource. Enter a value if you want to update it. | String | No |

**Example Payload**

```
Input:
{
"resourceCode" : "LBR-MOLD28",
"organizationCode" : "PR1",
"newResourceCode" : "LBR-MOLD29",
"disableDate" : "2017-12-14T14:00:00.000Z"
}

Output:
{
"status": "success"
}
```

## Find Resource (getresource)

URL: <host:>aimfgapi/vof/v1/fsm/setup/common/getresource

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |

**Example Payload**

```
Input
{
   "resourceCode" : "LBR-MOLD30",
    "organizationCode" : "PR1"
}

Output:

{
   "description":"Molding Labor",
   "disableDate":"2017-12-12T14:00:00.000Z",
   "organizationCode":"PR1",
   "organizationId":1381,
   "resourceCode":"LBR-MOLD30",
   "resourceId":-21,
   "resourceType":2,
   "status":"success"
}
```

## Create Resource-Employee Association (createresempassoc)

URL: <host:>/aimfgapi/vof/v1/fsm/setup/common/createresempassoc

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| personId | Person ID. | Integer | Yes |
| effectiveStartDate | Effective start date for the association. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ" | Date | Yes |
| effectiveEndDate | Effective end date for the association. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ" | Date | Yes |
| instanceId | This output parameter provides the ID for the person to resource association. | Integer | This is an output parameter. |

**Example Payload**

```
Input:
{
 "resourceId" : -1081,
    "organizationId" :1381,
    "personId" : 30750 ,
    "effectiveStartDate" : "2016-12-12T14:00:00.000Z",
    "effevtiveEndDate" : "2019-12-12T14:00:00.000Z"
}

Output:
{
    "instanceId": -601,
    "status": "success"
}
```

## Update Resource-Employee Association (changeresempassoc)

URL: <host:>/aimfgapi/vof/v1/fsm/setup/common/changeresempassoc

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| personId | Person ID. | Integer | Yes |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| effectiveStartDate | Effective start date for the association. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ" | Date | No |
| effectiveEndDate | Effective end date for the association. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ" | Date | No |

### Example Payload

```
Input:
{
   "resourceId":-1081,
    "personId" :30750 ,
    "effectiveStartDate" : "2016-12-12T14:00:00.000Z",
    "effevtiveEndDate":"2018-12-12T14:00:00.000Z"
}

Output:
{
    "status": "success"
}
```

## Find Resource-Employee Association (getinstanceId)

URL: <host:>/aimfgapi/vof/v1/fsm/setup/common/getinstanceId

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| personId | Person ID. | Integer | Yes |
| instanceId | ID for the person to resource association. | Integer | Yes, if both personId & Resource details are not provided. |

**Example Payload**

```
Input:
{

     "resourceId" : -1081,
     "personId" :30750
   }

Output:
{
    "effectiveStartDate": "2016-12-12T14:00:00.000Z",
    "effevtiveEndDate": "2018-12-12T14:00:00.000Z",
    "instanceId": -601,
    "organizationId": 1381,
    "personId": 30750,
    "resourceId": -1081,
    "status": "success"
}
```

## Remove Resource-Employee Association (removeresempassoc)

URL: <host:>/aimfgapi/vof/v1/fsm/setup/common/removeresempassoc

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| personId | Person ID. | Integer | Yes |

**Example Payload**

```
Input:
{
   "resourceId":-1081,
     "personId":30750
   }
Output:
{
    "status": "success"
}
```

## Employee Job Assignment (createempjobassignment)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/createempjobassignment

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| workOrderId | Work order ID. | Integer | A value is required for either workOrderId or workOrder. |
| workOrder | Work order name. | String | A value is required for either workOrderId or workOrder. |
| operationSeqNumber | Operation sequence number. | Integer | Yes |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| startDate | Job assignment start date. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | Yes |
| completionDate | Job assignment end date. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | Yes |
| instanceId | Instance ID generated during employee resource association. | Integer | Yes |

**Example Payload**

```
Input:
{
    "workOrder" : "233160",
    "operationSeqNumber" : 10,
    "organizationCode" : "PR2",
    "startDate" : "2016-07-22T14:00:00.000Z",
    "completionDate" : "2016-07-22T18:00:00.000Z",
    "instanceId":-601
}

Output:
{
    "status": "Success"
}
```

## Change/Update Employee Job Assignment (changeempjobassignment)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/changeempjobassignment

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| workOrderId | Work order ID. | Integer | A value is required for either workOrderId or workOrder. |
| workOrder | Work order name. | String | A value is required for either workOrderId or workOrder. |
| operationSeqNumber | Operation sequence number. | Integer | Yes |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| startDate | Job assignment start date. Enter if you want to update this value. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | No |
| completionDate | Job assignment end date. Enter if you want to update this value. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | No |
| instanceId | Instance ID generated during employee resource association. | String | Yes |

**Example Payload**

```
Input:

{
    "workOrder" : "233160",
    "operationSeqNumber" : 10,
    "organizationCode" : "PR2",
    "startDate" : "2016-07-23T14:00:00.000Z",
    "completionDate" : "2016-07-23T18:00:00.000Z",
    "instanceId":-601
}
output:
{
    "status": "Success"
}
```

## Find Jobs Assigned to Employee (getempassignmentforjob)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/getempassignmentforjob

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| startDate | Start date of date range to search for job assignments. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | Yes |
| completionDate | End date of date range to search for job assignments. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | Yes |
| instanceId | Instance ID generated during employee resource association. | Integer | Yes |

### Example Payload

```
Input:
{
    "organizationId" : 207,
    "startDate" : "2016-07-23T12:00:00.000Z",
    "completionDate" : "2016-07-23T19:00:00.000Z",
    "instanceId":-601
}

Output:
{
"count":1,
"items":[
{
"type":"rsrcInstanceListRow",
"rsrcInstanceDataRow":
 [
    {
"instanceId":-601,
"operationSeqNumber":10,
"organizationId":1382,
"resourceSeqnumber":0,
"workOrder":"233160",
"workOrderId":2712172
   }
 ],
"status":"Success"}],
"totalResults":1
}
```

## Find Employees Assigned to a Job (getjobassignmentforemp)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/getjobassignmentforemp

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| workOrderId | Work order ID. | Integer | A value is required for either workOrderId or workOrder. |
| workOrder | Work order name. | String | A value is required for either workOrderId or workOrder. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| startDate | Start date of date range to search for job assignments. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | No |
| completionDate | End date of date range to search for job assignments. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | No |

### Example Payload

```
Input:

{
    "workOrder" : "233160",
    "operationSeqNumber" : 10,
    "organizationCode" : "PR2",
    "startDate" : "2016-07-23T12:00:00.000Z",
    "completionDate" : "2016-07-23T19:00:00.000Z"
}

Output:
{
"count":1,
"items":[
{
"type":"rsrcInstanceListRow",
"rsrcInstanceDataRow":
 [
    {
"instanceId":-601,
"operationSeqNumber":10,
"organizationId":1382,
"resourceSeqnumber":0,
"workOrder":"233160",
"workOrderId":2712172
   }
 ],
"status":"Success"}],
"totalResults":1
}
```

## Resource Equipment Association (createresequpassoc)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/createresequpassoc

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| inventoryItemId | The item ID of the inventory item marked as an equipment item in Inventory. | Integer | A value is required for either inventoryItemId or inventoryItemName. |
| inventoryItemName | The item name of the inventory item marked as an equipment item in Inventory. | String | A value is required for either inventoryItemId or inventoryItemName. |

**Example**
**Example Payload**

```
Input:

{

   "resourceId": -1082,
    "equipmentId": 710045,
      "organizationCode": "PR1"
}
Output:
{
    "status": "success"
}
```

## Update Resource Equipment Association (updatersrcequipment)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/updatersrcequipment

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| inventoryItemId | The item ID of the inventory item marked as an equipment item in Inventory. Enter a value if you want to update this. | Integer | A value is required for either inventoryItemId or inventoryItemName. |
| inventoryItemName | The item name of the inventory item marked as an equipment item in Inventory. Enter a value if you want to update this. | String | A value is required for either inventoryItemId or inventoryItemName. |

**Example**
**Example Payload**

```
Input:
{

   "resourceId": -1082,
   "equipmentId": 721043,
   "organizationCode": "PR1"
}

Output:
{
   "status": "success"
}
```

## Find Resource Equipment Association (getreseqpassoc)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/getreseqpassoc

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| inventoryItemId | The item ID of the inventory item marked as an equipment item in Inventory. | Integer | A value is required for either inventoryItemId or inventoryItemName. |
| inventoryItemName | The item name of the inventory item marked as an equipment item in Inventory. | String | A value is required for either inventoryItemId or inventoryItemName. |

**Example**
**Example Payload**

```
Input:

{
   "resourceId":-1082,
   "equipmentId":721043,
    "organizationCode":"PR1"
}

Output:
{
    "count": 1,
    "items": [
        {
            "type": "rsrcEquipmentDetailsRow",
            "rsrcEquipmentDataRow": [
                {
                    "equipmentId": 721043,
                    "organizationId": 1381,
                    "resourceId": -1082
                }
            ],
            "status": "Success"
        }
    ],
    "totalResults": 1
}
```

## Remove Resource Equipment Association (removeresequipassoc)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/removeresequipassoc

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| inventoryItemId | The item ID of the inventory item marked as an equipment item in Inventory. | Integer | A value is required for either inventoryItemId or inventoryItemName. |
| inventoryItemName | The item name of the inventory item marked as an equipment item in Inventory. | String | A value is required for either inventoryItemId or inventoryItemName. |

**Example**
**Example Payload**

```
Input:
{
   "resourceId": -1082,
   "equipmentId": 721043,
   "organizationCode": "PR1"
}

Output:
{
   "status": "success"
}
```

## Log Resource In Time (logresourceintime)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/logresourceintime

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| workOrderId | Work order ID. Value for this field is considered for clock-in. | Integer | No, but if a value is passed, it is treated as a clock in to the Work Order. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| workOrder | Work order name. Value for this field is considered for clock-in. | String | No, but if a value is passed, it is treated as a clock in to the Work Order. |
| operationSeqNumber | Operation sequence number. | Integer | Required if Work Order detail is provided. |
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| serialNumber | Serial number. | String | No |
| assemblySerialNumber | Assembly Serial number. | String | No |
| statusType | Value must be always 1. | Integer | Yes |
| startDate | Date and time of shift-in or clock-in. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | Yes |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| endDate | Date and time of shift-out or clock-out. Do not provide a parameter value if the operator is currently shifted-in or clocked in. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | No |
| employeeId | Person ID entered while associating an employee to a resource. | String | Yes |

**Example Payload**

```
Input for Shift-In:
{
    "organizationId" : 1382,
    "startDate" : "2018-03-08T14:00:00.000Z",
     "statusType" : 1,
    "employeeId" : 3527,
     "resourceId" : -5
}

  Input for Clock-In:

  {
    "organizationId" : 1382,
    "startDate" : "2018-03-08T14:00:00.000Z",
     "statusType" : 1,
    "employeeId" : 3527,
    "workOrder" : 233159,
    "operationSeqNumber" : 20,
     "resourceId" : -5


  }

Output:
{
    "status": "Success",
    "timeEntryId": -1101
}
```

## Log Resource Out Time (logresourceouttime)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/logresourceouttime

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| timeEntryId | Time entry ID generated and is an output of the Log Resource In Time API (logresourceintime) at the time of shift-in or clock-in of employee. | Integer | Either pass only this parameter or pass the remaining set of parameters. |
| workOrderId | Work order ID. Value for this field is considered for clock-in. | Integer | No, but if a value is passed, it is treated as a clock out to the Work Order. |
| workOrder | Work order name. Value for this field is considered for clock-in. | String | No, but if a value is passed, it is treated as a clock out to the Work Order. |
| operationSeqNumber | Operation sequence number. | Integer | Required if Work Order detail is provided. |
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | A value is required for either organizationCode or OrganizationId. |
| organizationCode | Organization code. | String | A value is required for either organizationCode or OrganizationId. |
| serialNumber | Serial number. | String | No |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| assemblySerialNumber | Assembly Serial number. | String | No |
| statusType | Value must be always 1. | Integer | Yes |
| startDate | Date and time of shift-in or clock-in. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | Yes |
| endDate | Date and time of shift-out or clock-out. Do not provide a parameter value if the operator is currently shifted-in or clocked in. Format: "yyyy-MM-ddTHH:mm:ss.SSSZ". | Date | No |
| employeeId | Person ID entered while associating an employee to a resource. | String | Yes |

**Example Payload**

```
Input for Shift-Out:
{
    "timeEntryId": -1101
}
OR

{
    "organizationId" : 1382,
    "endDate" : "2018-03-08T20:00:00.000Z",
     "statusType" : 1,
    "employeeId" : 3527,
     "resourceId" : -5
}

  Input for Clock-Out:
{
    "timeEntryId": -1101
}

OR

  {
    "organizationId" : 1382,
    "endDate " : "2018-03-08T20:00:00.000Z",
     "statusType" : 1,
    "employeeId" : 3527,
    "workOrder" : 233159,
    "operationSeqNumber" : 20,
     "resourceId" : -5
  }

Output:
{
    "status": "Success"
}
```

## Find Employee Shift-In Details (isresshiftedin)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/isresshiftedin

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| shiftIn | This is an output parameter. Indicates if the employee is shifted-in. | String | This is an output field, not an input parameter. |
| timeEntryId | This Time entry ID is an output parameter, which is generated at the time of shift-in. | Integer | This is an output field, not an input parameter. |

### Example Payload

```
Input:
{
   "resourceId" : -22
}

Output:
{
    "employeeId": 4090,
    "instanceId": 0,
    "organizationId": 0,
    "shiftIn": "YES",
    "startDate": "2018-03-08T14:00:00.000Z",
    "status": "success",
    "statusType": 1,
    "timeEntryId": -25,
    "timeEntryMode": 8
}
```

## Find Employee Clock-In Details (isresshiftedin)

URL: <host>/aimfgapi/vof/v1/fsm/setup/common/isresclockedin

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceId | Resource ID. | Integer | A value is required for either resourceId or resourceCode. |
| resourceCode | Resource code. | String | A value is required for either resourceId or resourceCode. |
| organizationId | Organization ID. | Integer | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| organizationCode | Organization code. | String | If resourceCode is provided instead of resourceId, a value is required for either organizationCode or organizationId. |
| message | This is an output parameter, which indicates if the employee is clocked-in. | String | This is an output field, not an input parameter. |
| timeEntryId | This Time entry ID is an output parameter, which is generated at the time of clock-in. | Integer | This is an output field, not an input parameter. |

### Example Payload

```
Input:
{
   "resourceId" : -22
}

Output:
  {
    "count": 1,
    "items": [
        {
            "type": "rscActTimesListRow",
            "datalist": [
                {
                    "employeeId": 4090,
                    "operationSeqNumber": 10,
                    "startDate": "08-MAR-18",
                    "statusType": 1,
                    "timeEntryId": -26,
                    "workOrderId": 903
                }
            ],
            "message": "Clocked In:YES",
            "status": "Success"
        }
    ],
    "totalResults": 1
}
```

## defineKPI

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/defineKPI

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category. | String | Yes |
| functionalArea | Functional Area. | String | Yes |
| valueType | Value type. | String | Yes |
| isVisible | Determines whether or not to display the custom KPI. | String | No. Valid values are Y or N. |
| kpiDisplaySeq | KPI display sequence. | Integer | No |

***Response Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category. | String | Yes |
| functionalArea | Functional area. | String | Yes |
| isVisible | Determines whether or not to display the custom KPI. | String | No |
| kpiDisplaySeq | KPI display sequence. | Integer | No |
| valueType | Value type. | String | No |
| status | Status | String | No |

### Example Payloads

```
Input:
{
 "paramName" : "Test Machine1",
 "category" : "Machine",
 "functionalArea" : "OPM",
 "isVisible" : "y",
 "kpiDisplaySeq" : 11,
 "valueType" : "AbsoluteNo"
}
Output:
{
 "category": "Machine",
 "functionalArea": "OPM",
 "isVisible": "y",
 "kpiDisplaySeq": 11,
 "paramName": "Test ",
 "status": "success",
 "valueType": "AbsoluteNo"
}
```

## defineKPIValue

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/defineKPIValue

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category. | String | Yes |
| orgCode | Organization code. | String | Yes |
| paramValue | Parameter value. | String | Yes |
| deptName | Department name. | String | Yes |
| classification | Classification. | String | Yes |

***Response Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category. | String | Yes |
| orgCode | Organization code. | String | Yes |
| paramValue | Parameter value. | String | Yes |
| deptName | Department name. | String | No |
| classification | Classification. | String | No |
| message | Message | String | No |
| status | Status | String | No |

### Example Payloads

```
Input:
{
 "paramName" : "Test Machine1",
 "category" : "Machine",
 "orgCode" : "M1",
 "paramValue" : 2,
 "deptName":"ASSEMBLY",
 "classification" : "Bad"
}

Output:
{
 "category": "Machine",
 "classification": "Bad",
 "deptName": "ASSEMBLY",
 "message": "defineKPIValue",
 "orgCode": "M1",
 "paramName": "Test Machine1",
 "paramValue": 2,
 "status": "success"
}
```

## updateKPI

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/updateKPI

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category | String | Yes |
| isVisible | Determines whether or not to display the updated KPI. | String | No. Valid values are Y or N. |
| kpiDisplaySeq | KPI display sequence. | Integer | No |
| functionalArea | Functional area. | String | No |
| valueType | Value type. | String | No |
| customParam | Custom parameter. | String | No |

***Response Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category. | String | Yes |
| isVisible | Determines whether or not to display the updated KPI. | String | No |
| kpiDisplaySeq | KPI display sequence. | Integer | No |
| status | Status. | String | No |

### Example Payloads

```
Input:
{
 "paramName" : "prediction_alerts",
 "category" : "Machine",
 "isVisible" : "Y",
 "kpiDisplaySeq" : 11
}
Output:
{
 "category": "Machine",
 "isVisible": "Y",
 "kpiDisplaySeq": 11,
 "paramName": "prediction_alerts",
 "status": "success"
}
```

## deleteKPI

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/deleteKPI

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category | String | Yes |

***Response Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramName | Parameter name. | String | Yes |
| category | Category | String | Yes |
| status | Status | String | No |

### Example Payloads

```
Input:
{
 "paramName" : "Test Machine1",
 "category" : "Machine"
}
Output:
{
 "category": "Machine",
 "paramName": "Test Machine1",
 "status": "success"
}
```

## fetchKPIList

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/fetchKPIList

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| type | Type. | String |
| category | Category. | String |
| customParam | Custom parameter. | String |
| functionalArea | Functional area. | String |
| isVisible | Determines whether or not to display the KPI. | String |
| kpiDisplaySeq | KPI display sequence. | Integer |
| paramId | Parameter ID. | Long |

| Parameter Name | Description | Type |
|---|---|---|
| paramName | Parameter name. | String |

**Example Response Payload**

```
Output:
{
    "count": 5,
    "items": [
        {
            "type": "fsmCustomKpiRow",
            "category": "Manpower",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 1,
            "paramId": 1,
            "paramName": "not_reported"
        },
        {
            "type": "fsmCustomKpiRow",
            "category": "Machine",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 1,
            "paramId": 8,
            "paramName": "down"
        },
        {
            "type": "fsmCustomKpiRow",
            "category": "Management",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 1,
            "paramId": 14,
            "paramName": "start_delayed"
        },
        {
            "type": "fsmCustomKpiRow",
            "category": "Material",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 1,
            "paramId": 23,
            "paramName": "shortage"
        },
        {
            "type": "fsmCustomKpiRow",
            "category": "Method",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 1,
            "paramId": 30,
            "paramName": "material_exceptions"
        }
    ],
    "totalResults": 5
}
```

## fetchKpisByCategory

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/fetchKpisByCategory/{machine}

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| type | Type. | String |
| category | Category. | String |
| customParam | Custom parameter. | String |
| functionalArea | Functional area. | String |
| isVisible | Determines whether or not to display the KPI. | String |
| kpiDisplaySeq | KPI display sequence. | Integer |
| paramId | Parameter ID. | Long |
| paramName | Parameter name. | String |

### Example Response Payload

```
Output:
{
    "count": 2,
    "items": [
        {
            "type": "fsmCustomKpiRow",
            "category": "Machine",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 1,
            "paramId": 8,
            "paramName": "down"
        },
        {
            "type": "fsmCustomKpiRow",
            "category": "Machine",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 2,
            "paramId": 9,
            "paramName": "job_unassigned"
        }
    ],
    "totalResults": 2
}
```

## fetchKpisByParamName

URL: <host>/aimfgapi/vof/v1/fsm/dashboard/kpi/fetchKpisByParamName/{prediction_alerts}

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| type | Type. | String |
| category | Category. | String |
| customParam | Custom parameter. | String |
| functionalArea | Functional area. | String |
| isVisible | Determines whether or not to display the KPI. | String |
| kpiDisplaySeq | KPI display sequence. | Integer |
| paramId | Parameter ID. | Long |
| paramName | Parameter name. | String |

**Example Response Payload**

```
Output:
{
    "count": 2,
    "items": [
        {
            "type": "fsmCustomKpiRow",
            "category": "Manpower",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 7,
            "paramId": 7,
            "paramName": "prediction_alerts"
        },
        {
            "type": "fsmCustomKpiRow",
            "category": "Management",
            "customParam": "N",
            "functionalArea": "COMMON",
            "isVisible": "Y",
            "kpiDisplaySeq": 9,
            "paramId": 22,
            "paramName": "prediction_alerts"
        }
    ],
    "totalResults": 2
}
```

## custompredictors (Predictor Get REST API)

URL: <host>/iotappscustompred/vof/v1/rta/modelbuilding/custompredictors

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| routstepCstmPredId | Routing step custom predictor ID. | Integer |
| organizationId | Organization ID. | Integer |
| inventoryItemId | Inventory item ID. | Integer |
| revisionId | Revision ID. | Integer |
| recipeId | Recipe ID. | Integer |
| recipeVersion | Recipe version. | Integer |
| routingId | Routing ID. | Integer |

| Parameter Name | Description | Type |
|---|---|---|
| routingRevision | Routing revision. | Integer |
| routingstepId | Routing step ID. | Integer |
| billSequenceId | Bill sequence ID. | Integer |
| billRevisionId | Bill revision ID. | Integer |
| oprnId | Operation ID. | Integer |
| enableSerialAnalysis | Enable serial analysis? Answer this question using a value of Y or N. | String |
| mapColumn | Map column. | String |
| paramCode | Parameter code. | String |
| paramValueCode | Parameter value code. | String |
| paramShortCodeType | Parameter short code type. | String |
| paramParentCode | Parameter parent code. | String |
| entity | Entity. | String |
| paramVarType | Parameter variable type. | String |
| paramCategory | Parameter category. | String |
| paramSubCategory | Parameter subcategory, which can be anything. Example: Custom Predictor | String |
| isBoolean | Is the parameter Boolean? Internal use. | String |
| feature lookup | Feature lookup. Internal use. | String |

**Example Response Payload**


```
{
    "count": 2,
    "responseData": [
        {
            "createdBy": "-1",
            "enableSerialAnalysis": "N",
            "item": "PROD1",
            "itemRevision": "A",
            "lastUpdateLogin": -1,
            "lastUpdatedBy": "-1",
            "mapColumn": "N_ATTR_1",
            "operation": 50,
            "orgCode": "PR1",
            "paramCategory": "METHOD",
            "paramCode": "A::A",
            "paramDataType": "NUMBER",
            "paramShortCodeType": "A",
            "paramSubCategory": "A",
            "paramValueCode": "A",
            "recipe": "PROD1",
            "recipeVersion": 55,
            "routstepCstmPredId": 11
        },
        {
            "bomRevision": "A",
            "bomType": "Primary",
            "createdBy": "-1",
            "enableSerialAnalysis": "Y",
            "item": "I10 CAR",
            "lastUpdateLogin": -1,
            "lastUpdatedBy": "-1",
            "mapColumn": "C_ATTR_4",
            "operation": 30,
            "orgCode": "M1",
            "paramCategory": "MATERIAL",
            "paramCode": "OPERATION::20 (VTH2)::DEPARTMENT::VTD001",
            "paramDataType": "CHARACTER",
            "paramShortCodeType": "NO_OP",
            "paramSubCategory": "OPERATOR",
            "paramValueCode": "OP_NO",
            "routingRevision": "A",
            "routingType": "Primary",
            "routstepCstmPredId": 1,
            "serialStartOp": 20
        }
    ],
    "returnMsg": "Success"
}
```


## custompredictors (Predictor Insert REST API)

URL: <host>/iotappscustompred/vof/v1/rta/modelbuilding/custompredictors

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| orgCode | Organization code. | String | Yes |
| item | Item name. | Integer | Yes |
| itemRevision | Item revision. | Integer | Yes, for Process organizations. No, for discrete organizations. |
| recipe | Recipe name. | Integer | Yes, for Process organizations. No, for discrete organizations. |
| recipeVersion | Recipe version. | Integer | Yes, for Process organizations. No, for discrete organizations. |
| routingDesignator | Routing designator. | Integer | Yes, for discrete organizations. No, for process organizations. |
| routingRevision | Routing revision. | Integer | Yes, for discrete organizations. No, for process organizations. |
| bomDesignator | Bill of material designator. | Integer | Yes, for Process organizations. No, for discrete organizations. |
| bomRevision | Bill of material revision. | Integer | Yes, for discrete organizations. No, for process organizations. |
| enableSerialAnalysis | Enable serial analysis? Answer this question using a value of Y or N. | String | Yes, for discrete organizations. No, for process organizations. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| serialStartOp | Serial start operation. | String | Yes, if enableSerialAnalysis is Y. |
| operation | Operation | String | Yes, for both process and discrete organizations. |
| mapColumn | Map column. | String | Yes. MapColumn value must be c_attr_1 to c_attr_450 or n_attr_1 to n_attr_450, depending on the data type. |
| paramCode | Parameter code. | String | Yes. The paramCode and ParamValueCode combination must be unique. |
| paramValueCode | Parameter value code. | String | Yes. The paramCode and paramValueCode combination must be unique. |
| paramShortCodeType | Parameter short code type. | String | Yes |
| paramParentCode | Parameter parent code. | String | No |
| entity | Not used currently. | String | No |
| paramDataType | Parameter data type. | String | Yes. Acceptable values are NUMBER or CHARACTER. |
| paramCategory | Parameter category. | String | Yes. Acceptable values are Man, Machine, Material, Method, or Management. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramSubCategory | Parameter subcategory, which can be anything. Example:Custom Predictor | String | Yes |
| isBoolean | For internal use only | String | No |
| featureLookupType | For internal use only | String | No |
| uomCode | Unit of measure code. | String | No |

### Process Manufacturing Example

When producing Strawberry Jam, you want to consider the supplier of Strawberries as a predictor for the data preparation context operation, 30.

```
{
      "orgCode": "PR1",
      "item": "Strawberry Jam",
      "itemRevision":"A",
       "recipe": "STRAWBERRY_JAM",
      "recipeVersion":1,
       "operation": 30,
      "mapColumn": "C_ATTR_40",
      "paramCategory": "MATERIAL",
      "paramSubCategory": "CUSTOM",
      "paramCode": "Item::Strawberry Jam",
      "paramParentCode": "Item::Strawberry Jam",
      "paramDataType": "CHARACTER",
      "paramShortCodeType": "SJ_SUPPLIER",
      "paramValueCode": "Material Supplier"

  }
```

### Discrete Manufacturing Example

When producing a Wind Turbine, you want to consider the vibration readings from the robot (ROBO) for the data preparation context operation, 30.

```
{
   "bomDesignator": "Primary",
   "bomRevision": "A",
   "item": "WT_5040",
   "mapColumn": "N_ATTR_1",
   "operation": 30,
   "orgCode": "M1",
   "paramCategory": "METHOD",
   "paramCode": "Custom_code::POWER",
   "paramDataType": "NUMBER",
   "paramParentCode": null,
   "paramShortCodeType": "CP_1",
   "paramSubCategory": "CUSTOM_SUB_CAT",
   "paramValueCode": "Custom Power Value",
   "routingDesignator": "Primary",
   "routingRevision": "A",
   "enableSerialAnalysis": "Y"
 }
```

## custompredictors (Predictor Update REST API)

URL: <host>/iotappscustompred/vof/v1/rta/modelbuilding/custompredictors/{routstepCstmPredId}

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| routstepCstmPredId | ID of the custom predictor to retrieve from the cloud service. Used in the request path. | Integer | Yes |
| mapColumn | Map column. | String | Yes. MapColumn value must be c_attr_1 to c_attr_450 or n_attr_1 to n_attr_450, depending on the data type |
| paramCode | Parameter code. | String | Yes. The paramCode and paramValueCode combination must be unique. |
| paramValueCode | Parameter value code. | String | Yes. The paramCode and paramValueCode combination must be unique. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| paramShortCodeTyp e | Parameter short code type. | String | Yes |
| paramParentCode | Parameter parent code. | String | No |
| entity | Entity. | String | No |
| paramDataType | Parameter data type. | String | Yes. Acceptable values are NUMBER or CHARACTER. |
| paramCategory | Parameter category. | String | Yes. Acceptable values are Man, Machine, Material, Method, or Management. |
| paramSubCategory | Parameter subcategory, which can be anything. Example:Custom Predictor | String | Yes |
| param_var_type | Parameter variable type. | String | No |
| uomCode | Unit of measure code. | String | No |

### Process Manufacturing Example

For the product Strawberry Jam, update the predictor, Material Supplier.

```
{
      "mapColumn": "C_ATTR_29",
      "paramShortCodeType": "SJAM_SUPPLIER",
      "paramValueCode": "Material Supplier"

  }
```

### Discrete Manufacturing Example

For the discrete serialized manufacturing product Wind Turbine, pdate the predictor, Custom Power Value.

```
   {
      "paramCode": "Custom_code::POWER_READING",
      "paramDataType": "NUMBER",
   }
```

## custompredictors (Predictor Delete REST API)

URL: <host>/iotappscustompred/vof/v1/rta/modelbuilding/custompredictors/{routstepCstmPredId}

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| routstepCstmPredId | ID of the custom predictor to retrieve from the cloud service. Used in the request path. | Integer | Yes |

**Response Output:** Record deleted successfully.

**Processing Validations:** Deletes a custom predictor if there is no dataset created for the context after the custom predictor definition for that context.

## custompredictors (Data Get REST API)

Choose one of the following web service URLs:

- To get a predictor: <host>/iotapps/vof/v1/rta/modelbuilding/custompredictors/{custPredId}
- To get all predictors: <host>/iotapps/vof/v1/rta/modelbuilding/custompredictors/
- To get all predictors for a certain organization: <host>/iotapps/vof/v1/rta/modelbuilding/custompredictors/{organization_code}

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| routstepCstmPredId | ID of the custom predictor to retrieve from the cloud service. Used in the request path. | Integer | Yes |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| Count | Number of predictor records retrieved. | Integer |
| responseData | Information about each predictor. | Object |
| returnMsg | Indicates status of the response, such as Success. | String |

**Get All Predictors: <host>/iotapps/vof/v1/rta/modelbuilding/custompredictors/**


```
{
  "count": 2,
  "responseData": [
    {
 "bomDesignator": "Primary",
      "bomRevision": "A",
      "item": "WT_5040",
      "mapColumn": "N_ATTR_1",
      "operation": 30,
      "orgCode": "M1",
      "paramCategory": "METHOD",
      "paramCode": "Custom_code::POWER",
      "paramDataType": "NUMBER",
      "paramParentCode": null,
      "paramShortCodeType": "CP_1",
      "paramSubCategory": "CUSTOM_SUB_CAT",
      "paramValueCode": "Custom Power Value",
      "routingDesignator": "Primary",
      "routingRevision": "A",
      "enableSerialAnalysis": "Y"
 "createdBy": "-1",
       "creationDate": null,
 "entity": null,
      "featureLookupType": null,
       "isBoolean": null,
       "isbinned": null,
       "lastUpdateDate": null,
       "lastUpdateLogin": -1,
       "lastUpdatedBy": "-1",
 "multirowFunc": null,
 "paramClass": null,
 "recipe": null,
 "recipeVersion": null,
       "requestId": null,
      "routstepCstmPredId": 100,
       "returnMessage": null,
      "target": null,
      "trasnformation": null,
       "uomCode": null

},
    {
    "orgCode": "PR1",
      "item": "Strawberry Jam",
      "itemRevision":"A",
       "recipe": "STRAWBERRY_JAM",
      "recipeVersion":1,
       "operation": 30,
      "mapColumn": "C_ATTR_40",
      "paramCategory": "MATERIAL",
      "paramSubCategory": "CUSTOM",
      "paramCode": "Item::Strawberry Jam",
      "paramParentCode": "Item::Strawberry Jam",
      "paramDataType": "CHARACTER",
      "paramShortCodeType": "SJ_SUPPLIER",
      "paramValueCode": "Material Supplier"
 "bomDesignator": "null",
      "bomRevision": "null",
      "createdBy": "-1",
      "creationDate": null,
      "enableSerialAnalysis": "null",
      "entity": null,
      "featureLookupType": null,
      "isBoolean": null,
      "isbinned": null,
```

```
"lastUpdateDate": null,
      "lastUpdateLogin": -1,
      "lastUpdatedBy": "-1",
      "paramClass": null,
      "pivotFunction": null,
    "returnMessage": null,
      "routingDesignator": "null",
      "routingRevision": "null",
      "routstepCstmPredId": 101,
      "scoringStep": null,
      "serialStartOp": 0,
      "target": null,
      "trasnformation": null,

    } ,
  "returnMsg": "Success"
}
```

## createcustomdatapredictors (Predictor Data Insert REST API)

This web service creates data for a predictor.

URL: <host>/iotappscustompred/vof/v1/rta/modelbuilding/createcustomdatapredictors

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| caseId | CaseId represents the unique batch/job for the analysis. It is the wip_entity_id from the wip_entities table. Used only for discrete organizations. | Integer | Yes, for discrete organizations. |
| batch_id | The batch ID from the gme_batch_header table. Used only for process organizations. | Integer | Yes, for process organizations. |
| CAttr1 to CAttr450 | The map column defined for the predictor. Choose from CAttr1 to CAttr450 if the predictor is a Character type. | String | Yes |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| NAttr1 to NAttr450 | The map column defined for the predictor. Choose from NAttr1 to NAttr450 if the predictor is a Number type. | Integer | Yes |

**Process Manufacturing Example**

For the Strawberry Jam product, insert the data for batch_id = 123 for predictor = Material Supplier.

```
{
      "caseId": 123,
      "CAttr29": "ALLEN SUPPLIER"

  }
```

**Discrete Manufacturing Example**

For the Wind Turbine serialized product, insert the data for work order = 234 for predictor = Custom Power Value.

```
{
           "caseId": 234,
        "NAttr1": 2.3

   }
```

## updatecustomdatapredictors (Predictor Data Update REST API)

This web service updates data for a predictor.

URL:
<host>/iotappscustompred/vof/v1/rta/modelbuilding/updatecustomdatapredictors/{caseId}

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| caseId | CaseId is the identifier of the custom predictor to retrieve from the cloud service. | Integer | Yes |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| CAttr1 to CAttr450 | The map column defined for the predictor. Choose from CAttr1 to CAttr450 if the predictor is a Character type. | String | Yes |
| NAttr1 to NAttr450 | The map column defined for the predictor. Choose from NAttr1 to NAttr450 if the predictor is a Number type. | Integer | Yes |

**Process Manufacturing Example**

For the Strawberry Jam product, insert the data for batch_id = 123 for predictor = Material Supplier.

```
{

      "caseId": 123,
      "CAttr29": "BERRY SUPPLIER"

  }
```

**Discrete Manufacturing Example**

For the Wind Turbine serialized product, insert the data for work order = 234 for predictor = Custom Power Value.

```
   {
            "caseId": 234,
        "NAttr1": 3

   }
```

## deletecustomdatapredictors (Predictor Data Delete REST API)

This web service deletes data for a predictor.

URL: <host>/iotappscustompred/vof/v1/rta/modelbuilding/deletecustomdatapredictors/{caseId}

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| caseId | CaseId is the identifier of the custom predictor to retrieve from the cloud service. | Integer | Yes |

## customdatapredictorslist (Data Get REST API)

Choose one of the following web service URLs:

- To get data for a predictor: <host>/iotapps/vof/v1/rta/modelbuilding/customdatapredictors/{caseId}
- To get data for all predictors: <host>/iotapps/vof/v1/rta/modelbuilding/customdatapredictorslist

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| caseId | ID of the custom predictor to retrieve from the cloud service. Used in the request path. | Integer | Yes |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| Count | Number of predictor records retrieved. | Integer |
| responseData | Information about each predictor. | Object |

| Parameter Name | Description | Type |
|---|---|---|
| returnMsg | Indicates status of the response, such as Success. | String |

### Example Output Response


```
{
   "count":5,
   "responseData":[
      {
         "CAttr4":"D",
         "NAttr4":4.0,
         "caseId":2934570,
         "createdby":"-1",
         "creationdate":"2018-02-23",
         "lastupdatedate":"2018-02-23",
         "lastupdatedby":"-1",
         "lastupdatelogin":-1
      },
      {
         "CAttr4":"D",
         "NAttr4":4.0,
         "caseId":2934571,
         "createdby":"-1",
         "creationdate":"2018-02-23",
         "lastupdatedate":"2018-02-23",
         "lastupdatedby":"-1",
         "lastupdatelogin":-1
      },
    {
         "CAttr1":"2",
         "caseId":1904169,
         "createdby":"-1",
         "creationdate":"2018-02-23",
         "lastupdatedate":"2018-02-23",
         "lastupdatedby":"-1",
         "lastupdatelogin":-1
      },
      {
         "CAttr1":"2",
         "caseId":1904170,
         "createdby":"-1",
         "creationdate":"2018-02-23",
         "lastupdatedate":"2018-02-23",
         "lastupdatedby":"-1",
         "lastupdatelogin":-1
      },
      {
         "CAttr1":"2",
         "caseId":1903503,
         "createdby":"-1",
         "creationdate":"2018-02-23",
         "lastupdatedate":"2018-02-23",
         "lastupdatedby":"-1",
         "lastupdatelogin":-1
      }
   ],
   "returnMsg":"Success"
}
```

## createTimeline (Work Order Timeline REST API)

URL: /aimfgapi/vof/v1/trace/timeline/createTimeline

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| entityType | Type of entity for which event is created. | Integer | Yes. The only valid value is 5. |
| entityName | Work order name. | String | Yes |
| organizationCode | Organization code. | String | Yes |
| eventDate | Date on which the event occurred, in ISO date format. | Integer | Yes. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| eventText | First line of event display. | String | Yes |
| eventText2 | Second line of event display. | String | Yes |
| displayType | The icon that is displayed for this event from the standard list. | String | No. Provide a valid icon mapping. See: Icon Mapping, page D-100. |
| activeStartDate | Date from which to show the event, in ISO date format. | Integer | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| activeEndDate | Date from which the event ends, in ISO date format. | Integer | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| modeFlag | INSERT - Create new event.<br>UPDATE - Update the existing event's active start or active end date. | String | Yes. Either INSERT or UPDATE is required. |
| textColor | Text color for display. | String | No. Supply a valid color hex code. |
| iconColor | Icon color | String | No. Supply a valid color hex code. |
| retainDetails | Determines whether to delete or retain the details when updating an event. N = remove details. Y = retain details. Null = same behavior as Y. | String | No. Valid values are Y, N, or null. |
| attributeText | Label in details. | String | No |
| attributeValue | Value in details. | String | No |
| sequenceNumber | Order of display in details. | Integer | No |
| textColor | Text color in details. | String | No. Supply a valid color hex code. |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| totalEvents | Total events. | Integer |
| validEvents | Valid events. | Integer |
| invalidEvents | Invalid events. | Integer |

### Example: Work Order Timeline Payload

```
{
    "events" : [
        {
            "entityName" : "130651",
            "entityType" : 5,
            "organizationCode" : "PR1",
            "eventDate" : "2017-12-11T09:57:19.000Z",
            "eventText" : "Tested",
   "eventText2" : "Verified",
            "displayType" : "5",
            "activeStartDate" : "2017-01-01T01:00:00.000Z",
            "activeEndDate" : "2018-02-02T01:00:00.000Z",
            "modeFlag" : "INSERT",
   "textColor" : "#67b460",
   "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Verified By",
                    "attributeValue" : "Martin ",
                    "sequenceNumber" : 1,
                    "highlight" : 2,
     "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "highlight" : 3,
     "textColor" : "#67b460"
                }
            ]
        },
        {
            "entityName" : "121994",
            "entityType" : 5,
            "organizationCode" : "M1",
            "eventDate" : "2017-12-11T01:00:00.000Z",
            "eventText" : "Terminated",
   "eventText2" : "Verified",
            "displayType" : "1",
            "activeStartDate" : "2017-03-03T01:00:00.000Z",
            "activeEndDate" : "2018-04-03T01:00:00.000Z",
            "modeFlag" : "INSERT",
   "textColor" : "#67b460",
   "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Terminated By",
                    "attributeValue" : "Mark Smith",
                    "sequenceNumber" : 1,
                    "highlight" : 2,
     "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "highlight" : 3,
     "textColor" : "#67b460"
                }
            ]
        },
  {
            "entityName" : "121995",
            "entityType" : 5,
            "organizationCode" : "M1",
```

```
"eventDate" : "2017-02-11T01:00:00.000Z",
            "eventText" : "Created",
   "eventText2" : "Verified",
            "displayType" : "1",
            "activeStartDate" : "2017-05-05T01:00:00.000Z",
            "activeEndDate" : "2018-06-06T01:00:00.000Z",
            "modeFlag" : "INSERT",
   "textColor" : "#67b460",
   "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Terminated By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1,
                    "highlight" : 2,
     "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "highlight" : 3,
     "textColor" : "#67b460"
                }
            ]
        }
    ]
}
```

**Example Output Responses**

This is an example of a successful output response:

```
{
    "events" : [
    ],
    "invalidEvents" : 0,
    "totalEvents" : 1,
    "validEvents" : 1
}
```

This is an example of an output response resulting in error:

```
{
    "events" : [
        {
            "activeEndDate" : "2018-06-06T01:00:00.000Z",
            "activeStartDate" : "2017-05-05T01:00:00.000Z",
            "details" : [
                {
                    "attributeText" : "Terminated By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1,
                    "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "textColor" : "#67b460"
                }
            ],
            "displayType" : "1",
            "entityName" : "121995",
            "entityType" : "21",
            "eventDate" : "2017-02-11T01:00:00.000Z",
            "eventText" : "Created",
            "iconColor" : "#67b460",
            "modeFlag" : "INSERT",
            "organizationCode" : "M1",
            "status" : 1,
            "statusCode" : "Invalid Entity Type",
            "textColor" : "#67b460"
        }
    ],
    "invalidEvents" : 1,
    "totalEvents" : 1,
    "validEvents" : 0
}
```


## createTimeline (Serial Timeline REST API)

URL: /aimfgapi/vof/v1/trace/timeline/createTimeline

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| entityType | Type of entity for which event is created. | Integer | Yes. The only valid value is 2. |
| serialNumber | Serial number. | String | Yes |
| itemName | Item name. | String | Yes |
| organizationCode | Organization code. | String | No |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| eventDate | Date on which the event occurred, in ISO date format. | Integer | Yes. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| eventText | First line of event display. | String | Yes |
| eventText2 | Second line of event display. | String | Yes |
| displayType | The icon that is displayed for this event from the standard list. | String | No. Provide a valid icon mapping. See: Icon Mapping, page D-100. |
| activeStartDate | Date from which to show the event, in ISO date format. | Integer | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| activeEndDate | Date from which the event ends, in ISO date format. | Integer | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| modeFlag | INSERT - Create new event.<br>UPDATE - Update the existing event's active start or active end date. | String | Yes. Either INSERT or UPDATE is required. |
| textColor | Text color for display. | String | No. Supply a valid color hex code. |
| iconColor | Icon color | String | No. Supply a valid color hex code. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| retainDetails | Determines whether to delete or retain the details when updating an event. N = remove details. Y = retain details. Null = same behavior as Y. | String | No. Valid values are Y, N, or null. |
| attributeText | Label in details. | String | No |
| attributeValue | Value in details. | String | No |
| sequenceNumber | Order of display in details. | Integer | No |
| textColor | Text color in details. | String | No. Supply a valid color hex code. |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| totalEvents | Total events. | Integer |
| validEvents | Valid events. | Integer |
| invalidEvents | Invalid events. | Integer |

### Example: Serial Timeline Payload

```
{
    "events" : [
        {
            "serialNumber" : "ARKS61797",
            "itemName" : "ARK PP Serial Item2",
            "entityType" : 2,
            "eventDate" : "2017-12-11T09:57:19.000Z",
            "eventText" : "Created",
    "eventText2" : "Serial Timeline event2",
    "organizationCode" : "M1",
            "displayType" : "1",
            "activeStartDate" : "2017-12-11T09:57:19.000Z",
            "activeEndDate" : "2018-12-11T09:57:19.000Z",
            "modeFlag" : "INSERT",
            "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Terminated By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2
                }
            ]
        }
    ]
}
```

### Example Output Responses

This is an example of a successful output response:

```
{
    "events" : [
    ],
    "invalidEvents" : 0,
    "totalEvents" : 1,
    "validEvents" : 1
}
```

This is an example of an output response resulting in error:

```
{
    "events" : [
        {
            "activeEndDate" : "2018-06-06T01:00:00.000Z",
            "activeStartDate" : "2017-05-05T01:00:00.000Z",
            "details" : [
                {
                    "attributeText" : "Terminated By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1,
                    "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "textColor" : "#67b460"
                }
            ],
            "displayType" : "1",
            "entityName" : "121995",
            "entityType" : "21",
            "eventDate" : "2017-02-11T01:00:00.000Z",
            "eventText" : "Created",
            "iconColor" : "#67b460",
            "modeFlag" : "INSERT",
            "organizationCode" : "M1",
            "status" : 1,
            "statusCode" : "Invalid Entity Type",
            "textColor" : "#67b460"
        }
    ],
    "invalidEvents" : 1,
    "totalEvents" : 1,
    "validEvents" : 0
}
```


## createTimeline (Equipment Timeline REST API)

URL: /aimfgapi/vof/v1/trace/timeline/createTimeline

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| entityType | Type of entity for which event is created. | Integer | Yes. The only valid value is 10. |
| departmentCode | Equipment department. Applicable only for discrete organizations. | String | Yes |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| resourceCode | Resource code for the equipment. | String | Yes |
| equipmentName | Equipment item name. | String | Yes |
| instanceNumber | Equipment instance number. Applicable only for process organizations. | Integer | Yes |
| serialNumber | Equipment Serial Number | String | Yes |
| organizationCode | Organization code. | String | No |
| eventDate | Date on which the event occurred, in ISO date format. | Integer | Yes. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| eventText | First line of event display. | String | Yes |
| eventText2 | Second line of event display. | String | Yes |
| displayType | The icon that is displayed for this event from the standard list. | String | No. Provide a valid icon mapping. See: Icon Mapping, page D-100. |
| activeStartDate | Date from which to show the event, in ISO date format. | Integer | No. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| activeEndDate | Date from which the event ends, in ISO date format. | Integer | No. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| modeFlag | INSERT - Create new event.<br>UPDATE - Update the existing event's active start or active end date. | String | Yes. Either INSERT or UPDATE is required. |
| textColor | Text color for display. | String | No. Supply a valid color hex code. |
| iconColor | Icon color | String | No. Supply a valid color hex code. |
| retainDetails | Determines whether to delete or retain the details when updating an event. N = remove details. Y = retain details. Null = same behavior as Y. | String | No. Valid values are Y, N, or null. |
| attributeText | Label in details. | String | No |
| attributeValue | Value in details. | String | No |
| sequenceNumber | Order of display in details. | Integer | No |
| textColor | Text color in details. | String | No. Supply a valid color hex code. |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| totalEvents | Total events. | Integer |
| validEvents | Valid events. | Integer |
| invalidEvents | Invalid events. | Integer |

### Example: Equipment Timeline Payload

```
/* For Process Enabled */

{
    "events" : [
        {
            "entityType" : 10,
            "instanceNumber" : 1,
            "resourceCode" : "1-BLENDER",
            "equipmentName" : "EQ-1002",
            "serialNumber" : "BLENDER-01",
            "organizationCode" : "PR1",
            "eventDate" : "2017-12-11T09:57:19.000Z",
            "eventText" : "Process Machine extended event 1",
    "eventText2" : "Process Machine extended event 2"
            "displayType" : 1,
            "activeStartDate" : "2017-12-11T09:57:19.000Z",
            "activeEndDate" : "2018-12-11T09:57:19.000Z",
            "modeFlag" : "INSERT",
            "textColor" : "#67b460",
            "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Started By ",
                    "attributeValue" : "Jonathan ",
                    "sequenceNumber" : 1,
                    "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Started Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "textColor" : "#67b460"
                }
            ]
        }
    ]
}

/* For Discrete */

{
    "events" : [
        {
            "entityType" : 10,
            "departmentCode" : "MOLDING",
            "resourceCode" : "INJECTMOLD",
            "equipmentName" : "EQ00001",
            "serialNumber" : "INJ00001",
            "organizationCode" : "M1",
            "eventDate" : "2017-12-11T09:57:19.000Z",
            "eventText" : "Discrete Machine extended event 1",
            "displayType" : 1,
            "activeStartDate" : "2017-12-11T09:57:19.000Z",
            "activeEndDate" : "2018-12-11T09:57:19.000Z",
            "modeFlag" : "INSERT",
            "textColor" : "#67b460",
            "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Started By ",
                    "attributeValue" : "Jonathan ",
                    "sequenceNumber" : 1,
                    "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Started Date",
```

```
"attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "textColor" : "#67b460"
                }
            ]
        }
    ]
}
```

**Example Output Responses**

This is an example of a successful output response:

```
{
    "events" : [
    ],
    "invalidEvents" : 0,
    "totalEvents" : 1,
    "validEvents" : 1
}
```

This is an example of an output response resulting in error:

```
{
    "events" : [
        {
            "activeEndDate" : "2018-06-06T01:00:00.000Z",
            "activeStartDate" : "2017-05-05T01:00:00.000Z",
            "details" : [
                {
                    "attributeText" : "Terminated By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1,
                    "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Termination Date",
                    "attributeValue" : "18-MAR-16",
                    "sequenceNumber" : 2,
                    "textColor" : "#67b460"
                }
            ],
            "displayType" : "1",
            "entityName" : "121995",
            "entityType" : "21",
            "eventDate" : "2017-02-11T01:00:00.000Z",
            "eventText" : "Created",
            "iconColor" : "#67b460",
            "modeFlag" : "INSERT",
            "organizationCode" : "M1",
            "status" : 1,
            "statusCode" : "Invalid Entity Type",
            "textColor" : "#67b460"
        }
    ],
    "invalidEvents" : 1,
    "totalEvents" : 1,
    "validEvents" : 0
}
```

## createTimeline (Lot Timeline REST API)

URL: /aimfgapi/vof/v1/trace/timeline/createTimeline

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| entityType | Type of entity for which event is created. | Integer | Yes. The only valid value is 1. |
| lotNumber | Lot number. | String | Yes |
| itemName | Item name. | String | Yes |
| organizationCode | Organization code. | String | Yes |
| eventDate | Date on which the event occurred, in ISO date format. | Integer | Yes. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| eventText | First line of event display. | String | Yes |
| eventText2 | Second line of event display. | String | Yes |
| displayType | The icon that is displayed for this event from the standard list. | String | No. Provide a valid icon mapping. See: Icon Mapping, page D-100. |
| activeStartDate | Date from which to show the event, in ISO date format. | Integer | No. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| activeEndDate | Date from which the event ends, in ISO date format. | Integer | No. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| modeFlag | INSERT - Create new event.<br>UPDATE - Update the existing event's active start or active end date. | String | Yes. Either INSERT or UPDATE is required. |
| textColor | Text color for display. | String | No. Supply a valid color hex code. |
| iconColor | Icon color | String | No. Supply a valid color hex code. |
| retainDetails | Determines whether to delete or retain the details when updating an event. N = remove details. Y = retain details. Null = same behavior as Y. | String | No. Valid values are Y, N, or null. |
| attributeText | Label in details. | String | No |
| attributeValue | Value in details. | String | No |
| sequenceNumber | Order of display in details. | Integer | No |
| textColor | Text color in details. | String | No. Supply a valid color hex code. |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| totalEvents | Total events. | Integer |
| validEvents | Valid events. | Integer |
| invalidEvents | Invalid events. | Integer |

**Example: Lot Timeline Payload**

```
{    "events" : [
        {
            "lotNumber" : "LT-ARKS61797",
            "itemName" : "ARK PP Item2",
            "entityType" : 1,
            "eventDate" : "2017-12-11T09:57:19.000Z",
            "eventText" : "Automation",
            "eventText2" : "Testing"
     "organizationCode" : "M1",
            "displayType" : "1",
            "activeStartDate" : "2017-12-11T09:57:19.000Z",
            "activeEndDate" : "2018-12-11T09:57:19.000Z",
            "modeFlag" : "INSERT",
            "iconColor" : "#67b460",
            "details" : [
                {
                    "attributeText" : "Tested By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1
                },
                {
                    "attributeText" : "Test Reason",
                    "attributeValue" : "Auditing",
                    "sequenceNumber" : 2
                }
            ]
        }
    ]
}
```

**Example Output Responses**

This is an example of a successful output response:

```
{
    "events" : [
    ],
    "invalidEvents" : 0,
    "totalEvents" : 1,
    "validEvents" : 1
}
```

This is an example of an output response resulting in error:

```
{
    "events" : [
{
            "activeEndDate" : "2018-06-06T01:00:00.000Z",
            "activeStartDate" : "2017-05-05T01:00:00.000Z",
            "details" : [
                {
                    "attributeText" : "Tested By",
                    "attributeValue" : "Jonathan Smith",
                    "sequenceNumber" : 1,
                    "textColor" : "#67b460"
                },
                {
                    "attributeText" : "Test Reason",
                    "attributeValue" : "Auditing",
                    "sequenceNumber" : 2,
                    "textColor" : "#67b460"
                }
            ],
            "displayType" : "1",
            "lotNumber" : "121995",
"itemName" : "121995",
            "entityType" : "1",
            "eventDate" : "2017-02-11T01:00:00.000Z",
            "eventText" : "Created",
            "iconColor" : "#67b460",
            "modeFlag" : "INSERT",
            "organizationCode" : "M1",
            "status" : 1,
            "statusCode" : "Invalid Entity Type",
            "textColor" : "#67b460"
        }
    ],
    "invalidEvents" : 1,
    "totalEvents" : 1,
    "validEvents" : 0
}
```


## insertUpdateEntity (User Defined Entity REST API)

URL: /aimfgapi/vof/v1/trace/setup/entity/insertUpdateEntity

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| entityType | Type of custom entity. | Integer | Yes. Any valid custom entity value of 100 or greater. |
| entityName | The name for the custom entity. | String | Yes. Can include up to 128 characters. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| StartDate | Effective start date for the association. | Date | No. Defaults to the system date. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| EndDate | Effective end date for the association. | Date | No. Defaults to the system date. ISO date format required ((yyyy-MM-dd'T'HH:mm:ss.SSSXXX)). |
| moduleName | The name of the module using the entity. | String | No |
| orderNumber | Display order in search. | Integer | No |
| isCustom | Identifies the entity as either custom or seeded. | Integer | No. Valid values are 1 (for seeded) or 2 (for custom). |
| displayType | Identifies the icon to display. | String | No. Valid values are 1 to 22. See: Icon Mapping, page D-100. |
| iconColor | Identifies the icon color of the entity. | String | No. The default value is #8dc592. Use a valid color hex code. |
| modeFlag | INSERT - Create new entity.<br>UPDATE - Update the existing entity's start or active date.<br>DELETE - Delete the entity. | String | Yes. A value of INSERT, DELETE, or UPDATE is required. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| filterLabel1 | Search filter label used when the entity is selected in the Timeline view. | String | No |
| filterLabel2 | Search filter label used when the entity is selected in the Timeline view. | String | No |
| filterLabel3 | Search filter label used when the entity is selected in the Timeline view. | String | No |
| viewType | Specifies whether entity can be viewed in the Timeline, Network, or both. | String | No. Valid values are T (for timeline), N (for network) or null (for both). Null is the default value. |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| totalEvents | Total events. | Integer |
| validEvents | Valid events. | Integer |
| invalidEvents | Invalid events. | Integer |

**Example: Insert an Entity Payload for a Timeline and Network**

```
{
"entities": [
{
"entityType": 101,
"entityName": "Quality_test",
"startDate": "2018-02-05T09:57:19.000Z",
"endDate" : "2018-02-05T09:57:19.000Z",
"moduleName" :"Manufacturing",
"orderNumber" :2,
"isCustom" :2,
"displayType":1,
"iconColor":"#fffff",
"filterLabel1":"Events",
"filterLabel2":"Operations",
"filterLabel3":"Alerts",
"modeFlag": "INSERT"
}
]
}
```

**Example: Insert an Entity Payload for a Timeline**

```
{
"entities": [
{
"entityType": 101,
"entityName": "Quality_test",
"startDate": "2018-02-05T09:57:19.000Z",
"endDate" : "2018-02-05T09:57:19.000Z",
"moduleName" :"Manufacturing",
"orderNumber" :2,
"isCustom" :2,
"displayType":1,
"iconColor":"#fffff",
"filterLabel1":"Events",
"filterLabel2":"Operations",
"filterLabel3":"Alerts",
"modeFlag": "INSERT"
"entityViews" : [
                {
                    "viewType" : "T"
                }
            ]
}
]
}
```

## createHeader (User Defined Entity Instance REST API)

URL: /aiamfgapi/vof/v1/trace/timeline/createHeader/

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| entityType | Type of entity for which event is created. | Integer | Yes. Any valid custom entity value of 100 or greater. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | Yes |
| subtype | Process or discrete organization. | Integer | Yes. Valid values are 1 or 2.<br>1 = Discrete, 2 = Process. |
| headerDisplayLine1 | Primary display name of the entity instance. | String | Yes |
| headerDisplayLine2 | Secondary display name of the entity instance. | String | Yes |
| headerDescription | Description of the entity instance. | String | Yes |
| modeFlag | INSERT - Create new entity.<br>UPDATE - Update the existing entity. | String | Yes. A value of INSERT or UPDATE is required. |
| entity_pk1 | Identifier1 for entity | String | Yes |
| entity_pk2 | Identifier2 for entity | String | Yes |
| entity_pk3 | Identifier3 for entity | String | Yes |
| entity_pk4 | Identifier4 for entity | String | Yes |
| entity_pk5 | Identifier5 for entity | String | Yes |

**Example: Insert a Header Payload**

```
{
"headers" : [
{
"entityType" : 201,
"organizationCode" :"PR1",
"subType":2,
"headerDisplayLine1" : "SR-00001",
"headerDisplayLine2" : "RNC SR",
"headerDescription" : "Internal Service Request created",
"modeFlag" : "INSERT",
"entity_pk1" : "RNC SOURCE",
"entity_pk2" : "RNC SEVERITY",
"entity_pk3" : "RNC STATUS",
"entity_pk4" : "RNC PRIORITY ",
"entity_pk5" : "RNC OWNER"
}
]
}
```

## createNetwork (Create Network)

URL: /aimfgapi/vof/v1/trace/network/createNetwork

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| **For the node structure:** | | | |
| objectType | Specifies the object type of the node. | Integer | Yes. Provide one of the following values: 1- Lot; 2-Serial; 5- Work Order; 6 - PO; 7- SO; 100 and above- User Defined. |
| ObjectId | Specifies the type of object ID. | Integer | No. Select the ObjectId based on the objectType value. 1- Lot GenobjectId; 2- Serial GenobjectId; 5- Work Order GenobjectId; 6 - PO Header Id; 7- SO Header Id; 100 and above- User Defined Object unique identifier. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| objectNumber | Specifies the type of number used for the object. | Integer | No. Select the objectNumber based on the ObjectType value. 1- Lot Number; 2-Serial Number; 5- Work Order ; 6 - PO Number; 7- SO Number; 100 and above- User Defined Object Name. |
| orgCode | Organization code for the object number specified. | String | Yes, but not required for objectType = 6 or 7. |
| itemName | The Inventory item associated with the object. | String | Yes, but not required for objectType = 6 or 7. |
| lineText1 | First line of node display. | String | No |
| lineText2 | Second line of node display. | String | No |
| lineText3 | Third line of node display. | String | No |
| textColor | The text color of the node display lines. | String | No, but use a valid color hex code. |
| iconColor | The node icon color. | String | No, but use a valid color hex code. |
| displayType | The icon that is displayed for this node. | | No. Select from the following valid values: 1- Lot; 2- Serial; 5- Work Order; 6 - PO; 7- SO; 100 and above- User Defined. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| modeFlag | Indicates whether to insert, update, or merge a node. | String | Yes. Valid values are: INSERT - inserts a new node. UPDATE - updates an existing node attribute. MERGE- updates node attributes on an existing node and creates nonexisting nodes. |
| retainDetails | Determines whether to delete or retain details during an event update. | String | No. Valid values are: N - details are removed. Y - details are retained. Null - same behavior as Y. |
| **Node Structure Details:** | | | |
| attributeText | Label in details. | String | No. |
| attributeValue | Value in details. | String | No. |
| sequenceNumber | Order of display. | String | No. |
| textColor | Text color. | String | No, but use a valid color hex code. |
| **Link Structure:** | | | |
| linkColor | Color of the line that links two nodes. | String | No, but use a valid color hex code. |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| modeFlag | Indicates whether to insert, update, or merge a link. | String | Yes. Valid values are: INSERT - inserts a new link. UPDATE - updates an existing link attribute. MERGE- updates attributes of an existing link and creates nonexisting links. |
| activeStartDate | Date from which to show the link. | | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| activeEndDate | Date from which to stop showing the link. | | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| retainDetails | Determines whether to delete or retain details when updating the event. | | No. Valid values are: N - details are removed. Y - details are retained. Null - same behavior as Y. |
| **Link Structure Details:** | | | |
| attributeText | Label in details. | String | No. |
| attributeValue | Value in details. | String | No. |
| sequenceNumber | Order of display. | String | No. |
| textColor | Text color. | String | No, but use a valid color hex code. |
| **For defining an indicator to a node:** | | | |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| indicatorCode | Code used to identify a user defined indicator icon. | Integer | No. Valid values are 1001 through 1006. |
| colorCode | Code used to identify the color of the user defined indicator icon. | Integer | No. Valid values are 1 and 2. 1 = Red and 2 = Green. |
| description | | String | No |
| startDate | Date from which to show the user defined indicator icon. | Date | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |
| endDate | Date from which to stop showing the user defined indicator icon. | Date | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| **For links:** | | |
| linksInserted | The number of new links created. | Integer |
| linksUpdated | The number of links changed. | Integer |
| totalLinks | The total number of inserted and updated links. | Integer |
| invalidLinks | The number of links with errors. | Integer |
| **For nodes:** | | |

| Parameter Name | Description | Type |
|---|---|---|
| nodesInserted | The number of new nodes created. | Integer |
| nodesUpdated | The number of nodes changed. | Integer |
| invalidNodes | The number of invalid nodes | Integer |
| totalNodes | The total number of inserted and updated nodes. | Integer |

**Example 1**

```
{
    "nodes" : [
        {
            "fromNode" : {
                "node" : {
                          "objectType" : 5,
                         "entityType" : 2
                          "objectNumber" : "BATCH100",
                          "orgCode" : "PR1",
                          "modeFlag" : "MERGE"
                      }
                 },
            "toNode" : {
                "node" : {
                         "objectType" : 112,
                         "objectNumber" : "10022",
                         "objectId" : 123,
                         "lineText1" : "Testing",
                         "lineText2" : "BatchAttr Test",
                         "lineText3" : "BatchAttr Test34 ",
                         "modeFlag" : "MERGE",
                         "retainDetails" : "Y",
                         "displayType" : "2",
                         "textColor":"#FF7F50",
                         "iconColor":"#FF7F50",
                          "details" : {
                             "detail" : [
                                 {
                                     "attributeText" :
"BatchCustNodeDtl",
                                     "attributeValue" : "Testing ",
                                     "sequenceNumber" : 1,
                                     "textColor" : "#9500d8"
                                  }
                             ]
                         }
                  }
            },
            "link" : {
                "linkColor" : "#0000FF",
                "modeFlag" : "MERGE",
                "activeStartDate" : "13-01-2017",
                "activeEndDate" : "13-01-2017",
                "retainDetails" : "N",
                "details" : {
                             "detail" : [
                                 {
                                     "attributeText" : "BatchAttrLink",
                                     "attributeValue" : "Testing ",
                                     "sequenceNumber" : 1,
                                     "textColor" : "#9500d8"
                                     }
                             ]
                         }
            }
        }
    ]
}
```

**Example 2**

```
{
    "nodes" : [
        {
            "fromNode" : {
                "node" : {
                         "objectType" : 7,
                         "objectNumber" : "150023",
                         "modeFlag" : "MERGE"
                       }
            },
            "toNode" : {
                "node" : {
                         "objectType" : 100,
                         "objectNumber" : "EM50021",
                         "lineText1" : "Bearing",
                         "lineText2" : "New Bearing",
                         "lineText3" : "Old #Bearing ",
                         "modeFlag" : "MERGE",
                         "retainDetails" : "Y"
                       }
            },
            "link" : {

                "modeFlag" : "MERGE",
               "activeStartDate" : "13-01-2017",
                "retainDetails" : "Y",
                "details" : {
                             "detail" : [
                                 {
                                     "attributeText" : "VendorDetails",
                                     "attributeValue" : "Vendor1Dtls ",
                                     "sequenceNumber" : 1,
                                     "textColor" : "#9500d8"
                                     }
                             ]
                           }
            }
        }
    ]
}
```

**Example 3**

```
{
    "nodes" : [
        {
            "fromNode" : {
                "node" : {
                         "objectType" : 7,
                         "objectNumber" : "150025",
                         "modeFlag" : "MERGE"

                        }
            },
            "toNode" : {
                "node" : {
                         "objectType" : 101,
                         "objectNumber" : "150025TurbineSerial",
                         "lineText1" : "My",
                         "lineText2" : "WindTurbine",
                         "lineText3" : "Down",
                         "modeFlag" : "MERGE",
                        "textColor":"#FF7F50",
                        "iconColor":"#FF7F50",
                         "retainDetails":"Y",
                         "details" : {
                             "detail" : [
                                 {
                                     "attributeText" : "InstalledBy",
                                     "attributeValue" : "Jonathan",
                                     "sequenceNumber" : 1,
                                     "textColor":"#FF7F50"
                                 }
                             ]
                         }
                        }
            },
            "link" : {
                "modeFlag" : "MERGE",
               "activeStartDate" : "13-01-2017",
                "activeEndDate" : "01-02-2017",
               "linkColor":"#FF7F50",
                "retainDetails" : "Y",
                "details" : {
                             "detail" : [
                                 {
                                     "attributeText" : "Shipped By",
                                     "attributeValue" : "Fleet",
                                     "sequenceNumber" : 1,
                                     "textColor" : "#FF7F50"
                                     }
                             ]
                         }
            }
        },
         {
            "fromNode" : {
                "node" : {
                         "objectType" : 102,
                         "objectNumber" : "TurbineIncidents2",
                         "lineText1" : "My",
                         "lineText2" : "Wind Turbine Incident2",
                         "lineText3" : "Rotor Problem",
                         "modeFlag" : "INSERT",
                        "textColor":"#FF7F50",
                        "iconColor":"#FF7F50",
                         "retainDetails":"Y",
                          "details" : {
```

```
"detail" : [
                                    {
                                        "attributeText" : "Reported By",
                                        "attributeValue" : "Jonathan",
                                        "sequenceNumber" : 1,
                                        "textColor":"#FF7F50"
                                    }
                                ]
                            }
                        }
                },
                "toNode" : {
                    "node" : {
                             "objectType" : 101,
                             "objectNumber" : "150025TurbineSerial",
                             "lineText1" : "My",
                             "lineText2" : "Wind Turbine",
                             "lineText3" : "Good",
                             "modeFlag" : "MERGE",
                            "textColor":"#FF7F50",
                            "iconColor":"#FF7F50",
                             "retainDetails":"Y",
                             "details" : {
                                 "detail" : [
                                     {
                                         "attributeText" : "Installed By",
                                         "attributeValue" : "Jonathan",
                                         "sequenceNumber" : 1,
                                         "textColor":"#FF7F50"
                                     }
                                 ]
                             }
                        }
                },
                "link" : {
                    "modeFlag" : "MERGE",
                   "activeStartDate" : "13-01-2017",

                   "linkColor":"#FF7F50",
                    "retainDetails" : "Y",
                    "details" : {
                                "detail" : [
                                    {
                                        "attributeText" : "Found By",
                                        "attributeValue" : "RPM Analysis",
                                        "sequenceNumber" : 1,
                                        "textColor" : "#FF7F50"
                                        },
                                         {
                                        "attributeText" : "Reported Date",
                                        "attributeValue" : "25-Jan-2017",
                                        "sequenceNumber" : 1,
                                        "textColor" : "#FF7F50"
                                        }
                                ]
                            }
                }
            }
        ]
}
```

### Example 4: Defining an Indicator to a Node

```
{
  "nodes": [{
    "fromNode": {
      "node": {
        "objectType": 100,
        "objectNumber": "EM50021",
        "lineText1": "Bearing",
        "lineText2": "New Bearing",
        "lineText3": "Old #Bearing ",
        "modeFlag": "INSERT",
        "retainDetails": "Y",
        "indicators": [{
        "indicatorCode": 1001,
        "colorCode": 1,
        "description": "description",
        "startDate": "2018-12-03T02:08:56.000Z",
        "endDate": "2019-11-29T16:30:00.000Z"
         }
        }
     }]
}
```

### Example Output Responses

This is an example of a successful output response:

```
{
    "links" : [ ],
    "linksInserted" : 1,
    "linksUpdated" : 0,
    "nodes" : [ ],
    "nodesInserted" : 2,
    "nodesUpdated" : 0,
    "totalLinks" : 1,
    "totalNodes" : 2
}
```

This is an example of an output response resulting in error:

```
{
    "invalidLinks" : 1,
    "links" : [
        {
            "failureReason" : "End Date should be greater than Start
Date",
            "status" : 2,
            "statusCode" : "INCORRECT_END_DATE",
            "activeEndDate" : "2017-04-09T02:08:56.000Z",
            "activeStartDate" : "2017-04-10T02:08:56.000Z"
        },
        {
            "failureReason" : "End Date should be greater than Start
Date",
            "status" : 2,
            "statusCode" : "INCORRECT_END_DATE",
            "activeEndDate" : "2017-04-09T02:08:56.000Z",
            "activeStartDate" : "2017-04-10T02:08:56.000Z"
        }
    ],
    "linksInserted" : 0,
    "linksUpdated" : 1,
    "nodes" : [
    ],
    "nodesInserted" : 0,
    "nodesUpdated" : 2,
    "totalLinks" : 1,
    "totalNodes" : 2
}
```


## Icon Mapping

The Seeded Icons table below lists the icons that display when using the corresponding display type in one of the above web services. Some of the icons are shown as black below, but are actually achromatic and superimposed on a preset background color. The background color indicates the type of material move transaction.

***Seeded Icons***

| Display Type | Icon Used | Description |
|---|---|---|
| 1 |  | Lot |
| 2 |  | Serial |

| Display Type | Icon Used | Description |
|---|---|---|
| 3 |  | Operation |
| 4 |  | Component |
| 5 |  | Work Order |
| 6 |  | Purchase Order |
| 7 |  | Shipment |
| 8 |  | Assembly |
| 9 |  | Exception |
| 10 |  | Operator |
| 11 |  | Equipment |

| Display Type | Icon Used | Description |
|---|---|---|
| 12 |  | Quality |
| 13 |  | Maintenance |
| 14 |  | Status Updates |
| 15 |  | Pattern Match |
| 16 |  | Threshold Violation |
| 17 |  | Material Movement Outside |
| 18 |  | Material Movement Inside |
| 19 |  | Return Procured Material |
| 20 |  | Movement Within the Organization |

| Display Type | Icon Used | Description |
|---|---|---|
| 21 |  | Movement Across the Organization |
| 22 |  | Customer Returns |
| Null/Default |  | Default Icon |

The User Defined Indicator Icons table below lists the icons that display when an indicator code is defined for a node.

***User Defined Indicator Icons***

| Indicator Code | Icon Used | Description |
|---|---|---|
| 1001 |   | Up/Down |
| 1002 |   | Yes/No |
| 1003 |   | Green/Red Flag |

| Indicator Code | Icon Used | Description |
|---|---|---|
| 1004 |   | Green/Red Signal |
| 1005 |   | Problem/Resolved |
| 1006 |   | Repair Up/Down |

# Using Outbound REST Web Services

## insights (Get Quality Insights)

URL: /aimfgapi/vof/v1/public/hda/quality/insights

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationId | Organization ID | String | Yes |
| itemName | Inventory item name | String | No |
| qualityElement | Quality element | String | No |
| insight | Insight | String | No |
| publishDate | Publish date | Date | No |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| count | Total number of quality insight records. | Integer |
| items | Container parameter for item details. | |
| **For items:** | | |
| type | Type of insight. | String |
| confidence | Confidence | Float/Double |
| insight | Insight name. | String |
| insightCode | Insight code. | String |
| insightId | Insight ID. | Integer |
| insightType | Type of insight. | String |
| itemDescription | Item description. | String |
| itemName | Item name. | String |
| kulc | Statistical measure. | Float/Double |
| lift | Statistical measure. | Float/Double |
| modelId | Insight model ID. | Integer |
| modelName | Insight model name. | String |
| operation | Operation name. | String |
| productRecipe | Product recipe. | String |
| publishDate | Model publish date. | Date |

| Parameter Name | Description | Type |
|---|---|---|
| qualityElement | Quality element. | String |
| routingStepNo | Routing step number. | Integer |
| ruleId | Rule ID. | Integer |
| serialManufactured | Is this item serial manufactured? Answered "Y" or "N". | String |
| support | The percentage of historical work orders supporting the pattern out of the total number of work orders analyzed. | Integer |
| text | Insight description. | String |
| total | Total number of insights found by the model. | Integer |
| workOrderEndDate | End date of work order date range. | Date |
| workOrderStartDate | Start date of work order date range. | Date |
| **For links:** | | |
| href | Web site link. | String |
| rel | Descriptive attribute to href. | String |

**Process Example: Get Quality Insights for Organization PR2 for Item AUTO02PROD1 between 2017-09-05T00:00:00.000Z and 2017-10-05T00:00:00.000Z**

Http://host:port/aimfgapi/vof/v1/public/hda/quality/insights?organizationCode=PR2&q=itemName=AUTO02PROD1,publishDate>=2017-09-05T00:00:00.000Z,publishDate<=2017-10-05T00:00:00.000Z

**Sample response payload:**

```
{"count":12,
"items":[
{"type":"hdaPubQualityInsightsRow",
"confidence":1.0,
"insight":"On Target",
"insightCode":"ON_TARGET",
"insightId":3271,
"insightType":"Quality",
"itemDescription":"AUTO02PROD1",
"itemName":"AUTO02PROD1",
"kulc":0.85,
"lift":0.35,
"modelId":3202,
"modelName":"AUTO02_SD_Insights1",
"operation":"50 (AUTO02OP5)",
"productRecipe":"AUTO02RC1-1",
"publishDate":"2017-09-06T23:52:20-07:00",
"qualityElement":"AUTO02-Test_Num",
"routingStepNo":"2766",
"ruleId":"631",
"serialManufactured":"N",
"support":0.35,
"text":"35%  of the work orders of AUTO02PROD1 had AUTO02-Test_Num at
Operation 50 (AUTO02OP5) On Target when  Operation 20 (AUTO02OP2)
started -14.5 Sec   or less after the completion of prior operation ",
"total":40,
"workOrderEndDate":"2017-09-07T23:59:59-07:00",
"workOrderStartDate":"2017-03-07T00:00:00-08:00"},
.
.
.],
"links":[
{"href":"http://host:port/aimfgapi/vof/v1/public/hda/quality/insights?
organizationCode=&organizationCode",
"rel":"self"},
{"href":"http://host:port/aimfgapi/vof/v1/public/hda/quality-insight-
search-form",
"rel":"search-form"}]}
```

**Discrete Example Response Payload**


```
{"count":6,

"items": [

{"type":
"hdaPubQualityInsightsRow",
"bomDesignator":"Primary",
"bomRevision":"A",
"confidence":0.8998330550918197,
"insight":"On Target",
"insightCode":"ON_TARGET",
"insightId":3903,
"insightType":"Quality",
"inventoryItemId":858167,
"itemDescription":"Pinion Gear for G15 Gear box",
"itemName":"G15-Pinion Gear 07",
"kulc":0.930308684408655,
"lift":0.0,
"modelId":3178,
"modelName":"RJ_PG_07_TM02",
"operation":"20",
"productRecipe":"G15-Pinion Gear 07-A",
"publishDate":"2017-09-19T21:57:17-07:00",
"qualityElement":"Tooth Depth",
"routing":"G15-Pinion Gear 07-A",
"routingStepNo":"658276",
"ruleId":"981",
"serialManufactured":"N",
"support":0.44954128440366975,
"text":"44.95% of the work orders of G15-Pinion Gear 07 had Tooth Depth
at Operation 20 On Target when AISI Grade of Component 3DP Main Mtl
was more than 1025 , average Internal Temperature of &RESOURCE during
the Operation 10-3DP was 120.86667 F or less, maximum Vibration
of &RESOURCE during the Operation 10-3DP was 35.5 Hz or less",
"total":1199,
"workOrderEndDate":"2015-08-10T23:59:59-07:00",
"workOrderStartDate":"2015-06-10T00:00:00-07:00"}
,

...

],

"links":[

{"rel": "self",

"href": "http://host:port/aimfgapi/vof/v1/public/hda/quality/insights?
organizationCode=&organizationCode"},

{"rel": "search-form",

"href": "http://host:port/aimfgapi/vof/v1/public/hda/insight-search-
form"}

]}
```


## quality-insight-search-form (Get Quality Insight Search Form)

URL: /aimfgapi/vof/v1/public/hda/quality-insight-search-form

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| code | Syntax used for an operator. | Boolean |
| types | Types of parameters. | String |
| description | Description of the operator code. | String |
| itemName | Item name. | string |
| publishDate | Insight publish date. | Date |
| operators | Alphanumerical code used for an operator. | String |
| rel | Descriptive attribute to href. | String |
| href | Website. | String |

**Example**


```
{
"syntax": {
  "operators": {
    "eq": {"code": ["="], "types": ["string", "integer", "boolean","
number"], "description": "equal", "example": "itemName=Item0933"},
    "ge": {"code": [">="], "types": ["date"], "description": "greater
than or equal to", "example": "publishDate>=2017-06-20T00:00:00.000Z"},
    "le": {"code": ["<="], "types": ["date"], "description": "less than
or equal to", "example": "publishDate<=2017-10-02T00:00:00.000Z"}
   },
  "logicalOperators": {
    "and": {"code": [","], "description": "logical and", "example":
"publishDate>=2017-06-20T00:00:00.000Z,publishDate<=2017-10-02T00:00:
00.000Z"}
  }
},
"propertyOperators": {
  "itemName": {
    "operators": ["eq"]
  },
    "publishDate": {
    "operators": ["ge"]
  },
    "publishDate": {
    "operators": ["le"]
  },
    "qualityElement": {
    "operators": ["eq"]
  },
    "insight": {
    "operators": ["eq"]
  }
},
"links": [
  {"rel": "self", "href": "http://host:
port/aimfgapi/vof/v1/public/hda/quality-insight-search-form"},
  {"rel": "search", "href": "http://host:
port/aimfgapi/vof/v1/public/hda/quality/insights"}
]
}
```


## insights (Get Yield Insights)

URL: /aimfgapi/vof/v1/public/hda/yield/insights

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | Yes |
| itemName | Inventory item ID. | String | No |
| targetMeasure | Time frame. | Integer | No |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| insight | Insight | String | No |
| publishDate | Insight publish date. | Date | No. ISO date format required (yyyy-MM-dd'T'HH:mm:ss.SSSXXX). |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| count | Total number of quality insight records. | Integer |
| items | Container parameter for item details. | |
| **For items:** | | |
| type | Type of insight. | String |
| bomDesignator | Bill of material designator. | String |
| bomRevision | Bill of material revision. | String |
| confidence | Confidence | Float/Double |
| insight | Insight name. | String |
| insightCode | Insight code. | String |
| insightId | Insight ID. | Integer |
| insightType | Type of insight. | String |
| itemDescription | Item description. | String |
| itemName | Item name. | String |

| Parameter Name | Description | Type |
|---|---|---|
| kulc | Statistical measure. | Float/Double |
| lift | Statistical measure. | Float/Double |
| modelId | Insight model ID. | Integer |
| modelName | Insight model name. | String |
| productRecipe | Product recipe. | String |
| publishDate | Model publish date. | Date |
| routing | Quality element. | String |
| routingStepNo | Routing step number. | Integer |
| ruleId | Rule ID. | Integer |
| serialManufactured | Is this item serial manufactured? Answered "Y" or "N". | String |
| support | The percentage of historical work orders supporting the pattern out of the total number of work orders analyzed. | Integer |
| targetMeasure | Target yield. | |
| text | Insight description. | String |
| total | Total number of insights found by the model. | Integer |
| workOrderEndDate | End date of work order date range. | Date |
| workOrderStartDate | Start date of work order date range. | Date |

| Parameter Name | Description | Type |
|---|---|---|
| **For links:** | | |
| href | Web site link. | String |
| rel | Descriptive attribute to href. | String |

**Sample Discrete Response Payload: Get Yield Insights for Organization M1 for Item MDP_PN_ASSM10 with target measure as WORKORDERYIELD**

URL: Http://host:port/aimfgapi/vof/v1/public/hda/yield/insights?organizationCode=M1&q=itemName=MDP_PN_ASSM10,targetMeasure=WORKORDERYIELD


```
{"count":24,
"items":[{"type":"hdaPubYieldInsightsRow",
"bomDesignator":"Primary",
"bomRevision":"A",
"confidence":1.0,
"insight":"Medium",
"insightCode":"MEDIUM",
"insightId":3413,
"insightType":"Yield",
"itemDescription":"Description for item 001",
"itemName":"MDP_PN_ASSM10",
"kulc":1.0,
"lift":0.0,
"modelId":3277,
"modelName":"MIXED_HDA4",
"productRecipe":"MDP_PN_ASSM10-A",
"publishDate":"2017-09-13T02:24:30-07:00",
"routing":"MDP_PN_ASSM10-A",
"routingStepNo":"902403",
"ruleId":"65125",
"serialManufactured":"N",
"support":0.25,
"targetMeasure":"WORKORDERYIELD",
"text":"25%  of the work orders for MDP_PN_ASSM10 had a Medium Yield
when  average BOF-DENSITY of &RESOURCE during the Operation 20-MDP2 was
between 25.2 and 30.9 HR  , quantity of MDP_PN10_COMP1 issued to
Operation 10-MDP1 was between 3 and 6 Ea ",
"total":40,
"workOrderEndDate":"2017-03-31T23:59:59-07:00",
"workOrderStartDate":"2017-01-30T00:00:00-08:00"},
.
.
.],
"links":[
{"href":"http://host:port/aimfgapi/vof/v1/public/hda/yield/insights?
organizationCode=&organizationCode",
"rel":"self"},
{"href":"http://host:port/aimfgapi/vof/v1/public/hda/yield-insight-
search-form",
"rel":"search-form"}]}
```

**Sample Process Response Payload**


```
{"count":6,

"items": [

{"type":"hdaPubYieldInsightsRow",
"confidence":0.723353293413174,
"insight":"Very High",
"insightCode":"VERY_HIGH",
"insightId":2641,
"insightType":"Yield",
"inventoryItemId":726055,
"itemDescription":"Hot Rolled Coil",
"itemName":"HR Coil",
"kulc":0.827726029422636,
"lift":0.591576885406464,
"modelId":2504,
"modelName":"SEED_HDA_TEST2",
"operation":"BATCHYIELD",
"productRecipe":"HRCL-3106-1",
"publishDate":"2017-04-07T08:52:21-07:00",
"routing":"NOVALUE",
"routingStepNo":"511",
"ruleId":"394656",
"support":0.591576885406464,
"targetMeasure":"BATCHYIELD",
"text":"59.16%  of the work orders for HR Coil had a Very High Yield
when  planned duration of 40 (MS-TAP) operation exceeded plan
by 3307.73% to 9305.4%  , duration of COOL activity at 70 (HRCL-
COOLSTRIP) operation
was 7 Min 45 Sec, duration of SAMPLE activity at 70 (HRCL-COOLSTRIP)
operation was 7 Min 45 Sec",
"total":1021,
"workOrderEndDate":"2015-08-01T23:59:59-07:00",
"workOrderStartDate":"2015-05-01T00:00:00-07:00"}
,

...

],

"links":[

{"rel": "self",

"href": "http://host:port/aimfgapi/vof/v1/public/hda/yield/insights?
organizationCode=&organizationCode"},

{"rel": "search-form",

"href": "http://host:port/aimfgapi/vof/v1/public/hda/insight-search-
form"}

]}
```


## yield-insight-search-form (Get Yield Insight Search Form)

URL: /aimfgapi/vof/v1/public/hda/yield-insight-search-form

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| code | Syntax used for an operator. | Boolean |
| types | Types of parameters. | String |
| description | Description of the operator code. | String |
| itemName | Item name. | string |
| publishDate | Insight publish date. | Date |
| operators | Alphanumerical code used for an operator. | String |
| targetMeasure | Target yield. | Integer |
| insight | Insight. | String |
| rel | Descriptive attribute to href. | String |
| href | Website. | String |

**Example**

Http://host:port/aimfgapi/vof/v1/public/hda/yield-insight-search-form

```
{
"syntax": {
  "operators": {
    "eq": {"code": ["="], "types": ["string", "integer", "boolean","
number"], "description": "equal", "example": "itemName=Item0933"},
    "ge": {"code": [">="], "types": ["date"], "description": "greater
than or equal to", "example": "publishDate>=2017-06-20T00:00:00.000Z"},
    "le": {"code": ["<="], "types": ["date"], "description": "less than
or equal to", "example": "publishDate<=2017-10-02T00:00:00.000Z"}
   },
  "logicalOperators": {
    "and": {"code": [","], "description": "logical and", "example":
"publishDate>=2017-06-20T00:00:00.000Z,publishDate<=2017-10-02T00:00:
00.000Z"}
  }
},
"propertyOperators": {
  "itemName": {
    "operators": ["eq"]
  },
    "publishDate": {
    "operators": ["ge"]
  },
    "publishDate": {
    "operators": ["le"]
  },
    "targetMeasure": {
    "operators": ["eq"]
  },
    "insight": {
    "operators": ["eq"]
  }
},
"links": [
  {"rel": "self", "href": "http://host:
port/aimfgapi/vof/v1/public/hda/yield-insight-search-form"},
  {"rel": "search", "href": "http://host:
port/aimfgapi/vof/v1/public/hda/yield/insights"}
]
}
```


## predictions (Get Predictions)

URL: /aimfgapi/vof/v1/public/rta/{kpiCode}/predictions

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | Yes |
| workOrder | Work order. | String | No |
| targetMeasure | An output target measure, such as work order yield. | String | No |

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| predictionClassification | Prediction classification. | String | No |
| status | Status. | String | No |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| count | Total number of prediction records. | Integer |
| items | Container parameter for item details. | |
| **For items:** | | |
| type | Type of prediction. | String |
| alertFlag | Alert flag | String |
| completedDuration | Number of hours taken to manufacture the item. | Float/Double |
| departmentCode | Department code. | String |
| item | Item name. | String |
| itemCategory | Item category. | String |
| itemDesc | Item description. | String |
| operation | Operation. | String |
| operationNo | Operation number. | Integer |
| operationStatusDesc | Operation status description. | String |
| organizationCode | Organization code. | String |

| Parameter Name | Description | Type |
|---|---|---|
| planCompletedDate | Planned completion date. | Date |
| planStartDate | Planned start date. | Date |
| plannedDuration | Number of hours planned to manufacture the item. | Float/Double |
| predictionClassification | Classification bin or range. | String |
| predictionId | Prediction ID. | Integer |
| status | Work order status. | String |
| targetMeasure | Target measure. | String |
| workOrder | Work order. | String |
| links | Container parameter for link details. | |
| **For links:** | | |
| href | Web site link. | String |
| rel | Descriptive attribute to href. | String |

### Example Process Organization Response Payload

URL:https://localhost:443/aimfgapi/vof/v1/public/rta/QUALITY/predictions?organizationCode=FP1

```
{"count":1,
"items":
  [{"type":"rtaPubPredictionsRow",
  "alertFlag":"N",
  "completedDuration":19166.4,
  "item":"Strawberry Jam LT item 001",
  "itemCategory":"Product",
  "itemDesc":"Bulk Processed Strawberry Jam",
  "operation":"AT_PRED_SJ_MODEL_01",
  "operationNo":"10",
  "operationStatusDesc":"IN_PROGRESS",
  "organizationCode":"FP1",
  "planCompletedDate":"2000-01-01",
  "planStartDate":"2000-01-01",
  "plannedDuration":0.0,
  "predictionClassification":"On Target",
  "predictionId":1991342,
  "status":"In Progress",
  "targetMeasure":"SS_SOLUBLE_SOLIDS",
  "workOrder":"SJ_001_-7222"}
  ],
"links":
  [{"href":"http://host:port/aimfgapi/vof/v1/public/rta/{kpiCode}
/predictions?organizationCode=organizationCode",
  "rel":"self"}]}
```


**Example Discrete Organization Response Payload**


```
{"count":22,
"items":
  [{"type":"rtaPubPredictionsRow",
  "alertFlag":"N",
  "completedDuration":0.0,
  "departmentCode":"CaseRecord",
  "item":"G15 LT Pinion Gear Data",
  "itemCategory":"Case Record Data",
  "itemDesc":"Pinion Gear Assembly",
  "operation":"TST",
  "operationNo":"20",
  "operationStatusDesc":"IN_PROGRESS",
  "organizationCode":"T11",
  "planCompletedDate":"2010-02-04",
  "planStartDate":"2010-02-04",
  "plannedDuration":2400.0,
  "predictionClassification":"On Target",
  "predictionId":1984327,
  "status":"In Progress",
  "targetMeasure":"TD_TOOTH_DEPTH",
  "workOrder":"AT_WOUP_2505336"},
  ...],
"links":
  [{"href":"http://host:port/aimfgapi/vof/v1/public/rta/{kpiCode}
/predictions?organizationCode=organizationCode",
  "rel":"self"}]}
```


## predictions/{predictionId: \d+} (Get Prediction Detail)

URL: /aimfgapi/vof/v1/public/rta/predictions/{predictionId: \d+}

***Request Parameters***

| Parameter Name | Description | Type | Required? |
|---|---|---|---|
| organizationCode | Organization code. | String | Yes |

***Response Parameters***

| Parameter Name | Description | Type |
|---|---|---|
| alert | Determines if this is a predictive alert. | String |
| classificationProbability | Container parameter for classification probability. | |
| **For classification probability:** | | |
| • binColor | Color of the classification bin or range. | String |
| • confidence | The accuracy of the model for the classification range. | Float/Double |
| • prediction | The classification range for the result. | String |
| classifications | Container parameter for classification bins. | String |
| **For classifications:** | | |
| • alert | Determines if this is a predictive alert. | String |
| • description | Alert description. | String |
| • prediction | The classification range for the result. | String |

| Parameter Name | Description | Type |
|---|---|---|
| • rangeEnd | The end of the classification range. | Float/Double |
| • rangeStart | The start of the classification range. | Float/Double |
| departmentCode | Department code. | String |
| itemDescription | Item description. | String |
| itemName | Item name. | String |
| links | Container parameter for link information. | |
| **For links:** | | |
| • href | Website. | String |
| • rel | Descriptive attribute to href. | String |
| operation | Operation. | String |
| operationNo | Operation number. | Integer |
| organizationCode | Organization code. | String |
| predictedMeasure | Predicted target measure. | String |
| predictionClassification | Classification bin or range. | String |
| predictionId | Prediction ID. | Integer |
| predictionModel | Prediction model name. | String |
| predictors | Container parameter for predictors. | |
| **For predictors:** | | |

| Parameter Name | Description | Type |
|---|---|---|
| • category | KPI category. | String |
| • id | KPI ID. | Integer |
| • name | KPI name. | String |
| • predictionInfluence | How much the KPI influences the target measure. | Float/Double |
| • rank | Rank of KPI. | Integer |
| • subCategory | KPI subcategory. | String |
| • value | KPI value. | String |
| • valueCode | KPI value code | String |
| probability | Probability percentage of true predictions. | Float/Double |
| rangeEnd | End of alert range. | Float/Double |
| rangeStart | Start of alert range. | Float/Double |
| timeline | Container parameter for the timeline. | |
| **For timeline:** | | |
| • binColor | Color of the classification bin or range. | String |
| • prediction | The classification range for the result. | String |
| • predictionDate | The date the prediction was made. | Date |

| Parameter Name | Description | Type |
|---|---|---|
| • predictionId | Prediction ID. | Integer |
| workOrder | Work order number. | String |
| workOrderCompleted | Indicates whether the work order completed. | String |
| workOrderStartDate | Work order start date. | Date |

**Example Discrete Organization Response Payload**

URL: https://hostport:443/aimfgapi/vof/v1/public/rta/predictions/1827693?organizationCode=T11

```
{"alert":false,
"classificationProbability":
  [{"binColor":"#309fdb","confidence":0.0,"prediction":"Within Lower
Limit"},{"binColor":"#ffb54d", "confidence":0.15730337078651685,"
prediction":"Within Upper Limit"},
  {"binColor":"#68c182","confidence":0.8426966292134831,"prediction":"On
Target"}],
"classifications":
  [{"alert":false,"description":null,"prediction":"Within Lower Limit","
rangeEnd":4,"rangeStart":4},
  {"alert":false,"description":null, "prediction":"On Target",
"rangeEnd":5,"rangeStart":4},
  {"alert":false,"description":null, "prediction":"Within Upper Limit",
"rangeEnd":6,"rangeStart":5}],
"departmentCode":"CaseRecord",
"itemDescription":"Pinion Gear Aseembly",
"itemName":"G15 LT Pinion Gear OOW Data",
"links":
  [{"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/1827693?
organizationCode=organizationCode","rel":"self"},
  {"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/2077495?
organizationCode=organizationCode","rel":"other-prediction-for-same-
work-order"},
  {"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/2079237?
organizationCode=organizationCode","rel":"TD_TOOTH_DEPTH-prediction-for-
similar-work-order"},
  {"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/2079238?
organizationCode=organizationCode","rel":"TD_TOOTH_DEPTH-prediction-for-
similar-work-order"},
  ...],
"operation":"TST",
"operationNo":"20",
"organizationCode":"T11",
"predictedMeasure":"TD_TOOTH_DEPTH",
"predictionClassification":"On Target",
"predictionId":1827693,
"predictionModel":"AT  Model 001",
"predictors":
  [{"category":"UNSPD","id":790,"name":"Y_SUPPLIER","
predictionInfluence":0.087,"rank":1,"subCategory":"Unspecified","
value":"NIMS Metals","valueCode":"Y_SUPPLIER"},
  {"category":"UNSPD","id":792,"name":"Y_AVERAGE_TEMPERATURE","
subCategory":"Unspecified","value":"117.8","valueCode":"
Y_AVERAGE_TEMPERATURE"},
  {"category":"UNSPD","id":793,"name":"Y_TENSILE_STRENGTH","
subCategory":"Unspecified","value":"65.78883","valueCode":"
Y_TENSILE_STRENGTH"},
  {"category":"UNSPD","id":791,"name":"Y_MAXIMUM_VIBRATION","
subCategory":"Unspecified","value":"35","valueCode":"
Y_MAXIMUM_VIBRATION"}],
"probability":0.8426966292134831,
"rangeEnd":5,
"rangeStart":4,
"timeline":
  [{"binColor":"#68c182","prediction":"On Target","predictionDate":"
2020-03-09T12:40:33.000Z","predictionId":1844304},
  {"binColor":"#68c182","prediction":"On Target","predictionDate":"2020-
03-09T13:10:33.000Z","predictionId":1846677},
  {"binColor":"#68c182","prediction":"On Target","predictionDate":"2020-
03-09T13:40:33.000Z","predictionId":1849050},
  {"binColor":"#68c182","prediction":"On Target","predictionDate":"2020-
03-09T14:10:33.000Z","predictionId":1851423},
```

```
...],
"workOrder":"AT_WOUP_2505336",
"workOrderCompleted":false,
"workOrderStartDate":"2010-02-04"}
```

**Example Process Organization Response Payload**


```
{"alert":false,
"classificationProbability":
  [{"binColor":"#ed6647","confidence":0.04316546762589928,"prediction":"
Above Upper Limit"},
  {"binColor":"#ffb54d","confidence":0.2805755395683453,"prediction":"
Within Upper Limit"},
  {"binColor":"#68c182","confidence":0.6762589928057554,"prediction":"On
Target"}],
"classifications":
  [{"alert":false,"description":null,"prediction":"On Target","
rangeEnd":63,"rangeStart":60},
  {"alert":false,"description":null,"prediction":"Within Upper Limit","
rangeEnd":66,"rangeStart":63},
  {"alert":false,"description":null,"prediction":"Above Upper Limit","
rangeEnd":69,"rangeStart":66}],
"itemDescription":"Bulk Processed Strawberry Jam",
"itemName":"Strawberry Jam LT item 001",
"links":
  [{"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/1827589?
organizationCode=organizationCode","rel":"self"},
  {"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/2106015?
organizationCode=organizationCode","rel":"other-prediction-for-same-
work-order"},
  {"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/2105247?
organizationCode=organizationCode","rel":"SS_SOLUBLE_SOLIDS-prediction-
for-similar-work-order"},
  {"href":"http://host:
port/aimfgapi/vof/v1/public/rta/predictions/2105248?
organizationCode=organizationCode","rel":"SS_SOLUBLE_SOLIDS-prediction-
for-similar-work-order"},
  ...]
"operation":"AT_PRED_SJ_MODEL_01",
"operationNo":"10",
"organizationCode":"FP1",
"predictedMeasure":"SS_SOLUBLE_SOLIDS",
"predictionClassification":"On Target",
"predictionId":1827589,
"predictionModel":"AT_PRED_FP1_SJ_M_2",
"predictors":
  [{"category":"UNSPD","id":424,"name":"SS_SJ_FRUIT_CALCIUM","
predictionInfluence":0.191,"rank":1,"subCategory":"Unspecified","
value":"21.45","valueCode":"SS_SJ_FRUIT_CALCIUM"},
  {"category":"UNSPD","id":423,"name":"SS_SJ_FRUIT_SOLUBLE_SOLIDS","
predictionInfluence":0.16,"rank":2,"subCategory":"Unspecified","value":"
63.09","valueCode":"SS_SJ_FRUIT_SOLUBLE_SOLIDS"},
  {"category":"UNSPD","id":425,"name":"SS_PECTIN_GRADE","
predictionInfluence":0.094,"rank":3,"subCategory":"Unspecified","
value":"Medium Set","valueCode":"SS_PECTIN_GRADE"},
  ...]
"probability":0.6762589928057554,
"rangeEnd":63,
"rangeStart":60,
"timeline":
  [{"binColor":"#68c182","prediction":"On Target","predictionDate":"
2020-03-09T18:06:16.000Z","predictionId":1870303},
  {"binColor":"#68c182","prediction":"On Target","predictionDate":"2020-
03-09T18:36:16.000Z","predictionId":1872676},
  {"binColor":"#68c182","prediction":"On Target","predictionDate":"2020-
03-09T19:06:16.000Z","predictionId":1875049},
  ...]
"workOrder":"SJ_001_-7222",
"workOrderCompleted":false,
```

```
"workOrderStartDate":"2000-01-01"}
```

# E

# Background Processes

This appendix covers the following topics:

- Running Background Processes

## Running Background Processes

You can select to schedule the programs to run at a scheduled time, repeating at a certain interval, or immediately. You can use the Background Process page to search for and manage background processes or submit a new request.

> **Important:** Only users with the Application Administrator role can submit a new request.

### To search for and manage background processes:

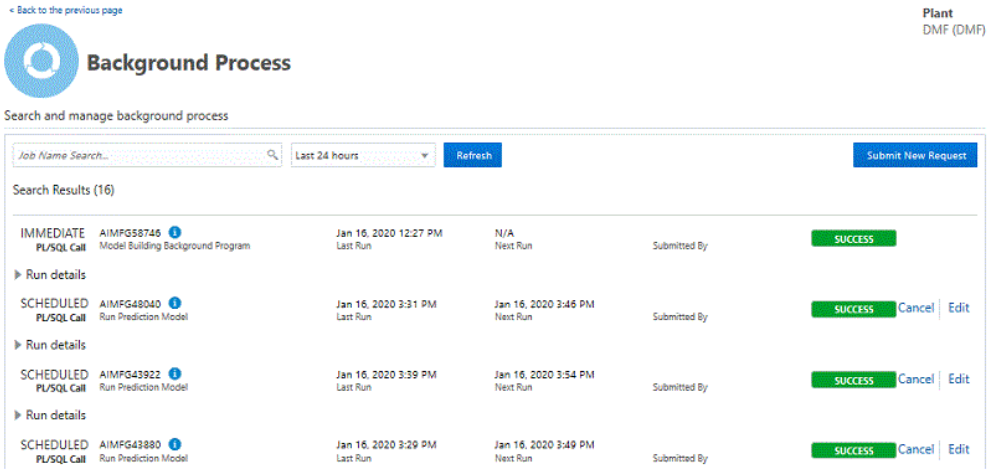
1. Navigate to the Background Process page.

   From the User Name drop-down list, select **Background Process**.

2. The Background Processes page appears with the list of programs that have been scheduled and run.

3. You can use the Job Name Search field to enter and search for a specific job for which a program has been run in the last 24 hours or last 7 days.

4. Click **Refresh**.




5. Use the Background Processes page to view the following information for each background process:

   - Immediate or scheduled type of run
   - Job Name
   - Program name
   - Date and time of last run
   - Date and time of next run

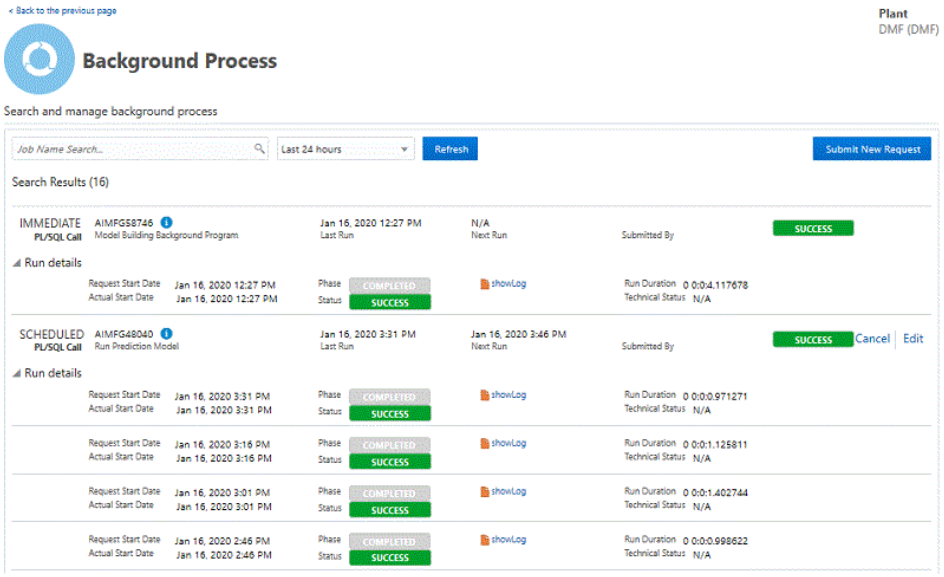
     **Optional Action:** If the job has a scheduled next run, you can click **Cancel** to cancel the scheduled job or **Edit** to change the repeat interval, start date, or end date.

   - Submitted By
   - Status of run

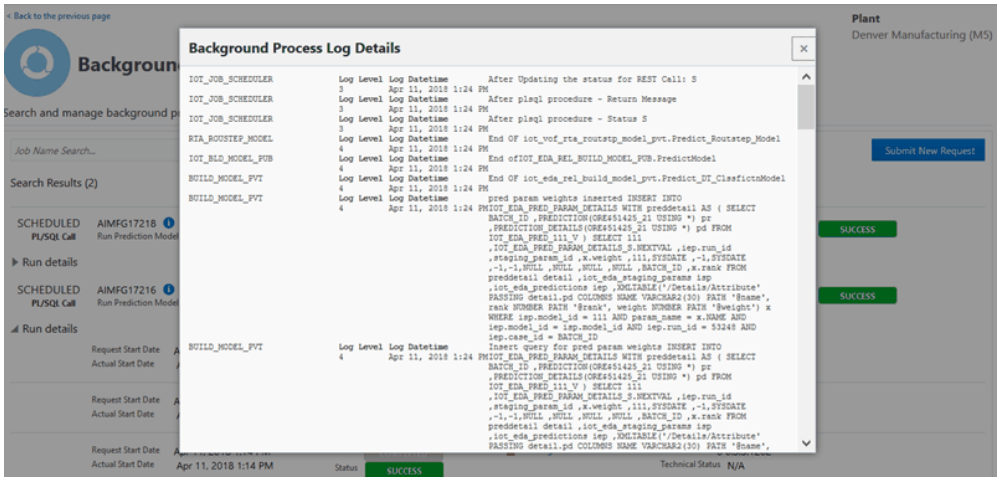
6. Click **Run details** to view:

   - Request Start Date
   - Actual Start Date
   - Phase
   - Log details

- Run Duration
- Technical Status

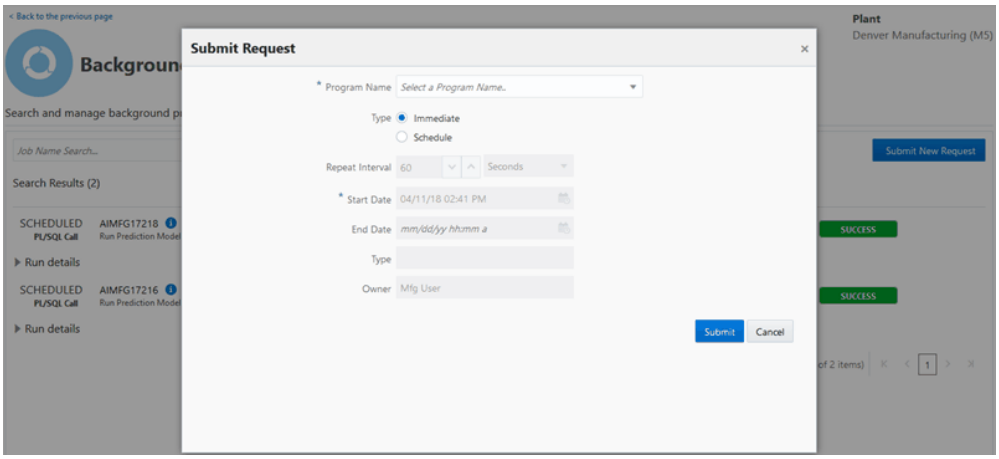



7. Click **showLog** to view Background Process Log Details.

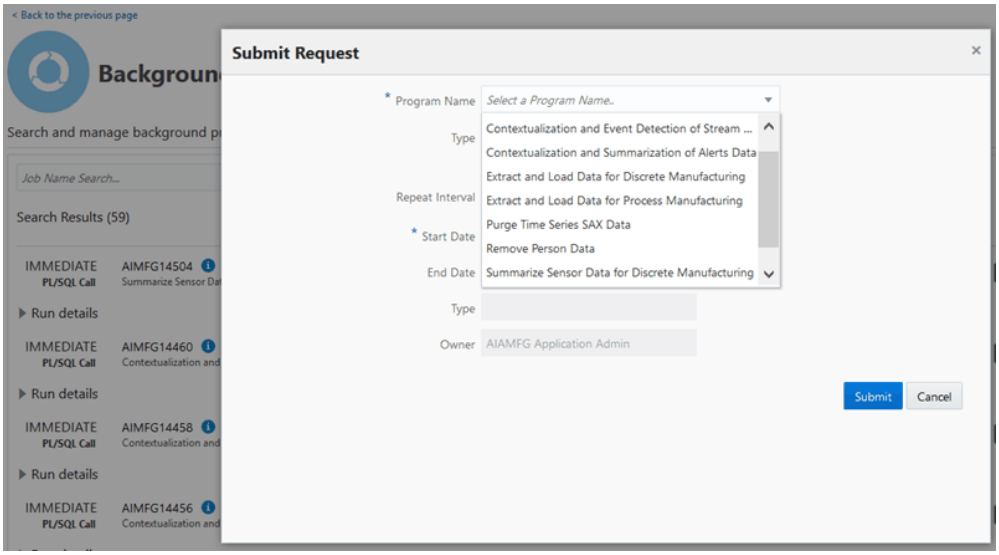



## To schedule and run a background process:

1. In the Background Process page, click **Submit New Request**.

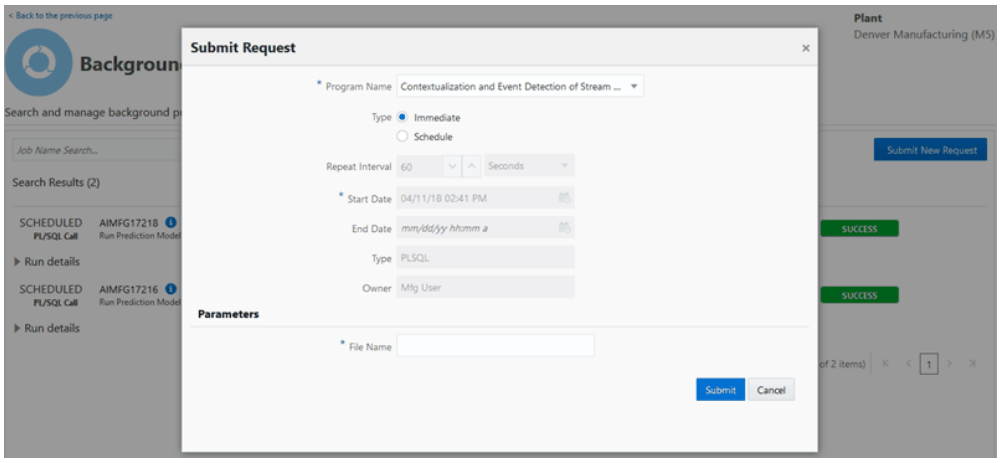



2. In the Submit Request page, select a program from the Program Name list.




3. Select Type as:
   - Immediate to run the program immediately.
   - Schedule to run the program at a later date and time.

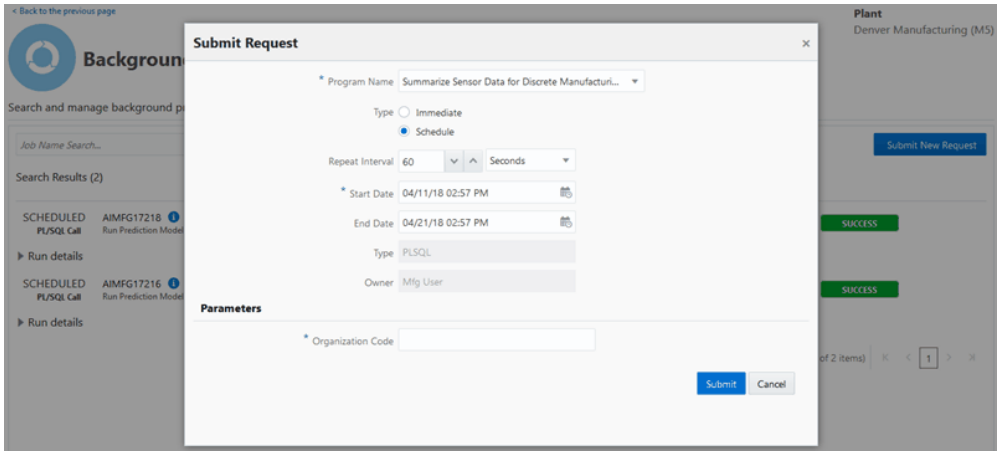
4. If you select Immediate, enter the parameters listed for the program you are running, and then click **Submit** to run the program.




5. If you select Schedule, enter details for the following:

   - **Repeat Interval**
   - **Start Date**
   - **End Date**

6. Enter the parameters listed for the program you are running.




7. Click **Submit** to schedule this request.

   The following table lists the background processes you can run in Adaptive Intelligent Apps for Manufacturing:

| Program | Description |
|---|---|

| | |
|---|---|
| Contextualization and Event Detection of Stream Data | Contextualizes stream data for an uploaded stream data file, with equipment context, work context and serial context, and derives the event from the stream data according to the setup of the machine event analysis feature set.<br><br>Enter the following parameter before submitting the program to run: File Name. |
| Contextualization and Summarization of Alerts Data | Contextualizes event/alert data for an uploaded alert data file with equipment context, work context, and serial context and summarizes the continual event/alert records with start and end time.<br><br>Enter the following parameter before submitting the program to run: File Name. |
| Extract and Load Data for Discrete Manufacturing | Retrieves the most recently loaded quality data for discrete manufacturing organizations from the data lake in the Oracle Cloud.<br><br>Enter the following parameter before submitting the program to run:<br><br>• Load Quality Data<br><br>• Load Yield Data<br><br>• Last Refresh Date |
| Extract and Load Data for Process Manufacturing | Retrieves the most recently loaded quality data for process manufacturing organizations from the data lake in the Oracle Cloud.<br><br>Enter the following parameter before submitting the program to run:<br><br>• Load Quality Data<br><br>• Load Yield Data<br><br>• Last Refresh Date |

| | |
|---|---|
| Maintain Multi-lingual Tables | Refreshes the translation of the translatable data ingested into AIAMFG. Run this process on an as-needed basis. |
| Purge Time Series SAX Data | Purges time series SAX data for a specified time period for a dataset.<br><br>Enter the following parameters before submitting the program to run:<br><br>• Dataset Name<br><br>• Dataset Creation Date From<br><br>• Dataset Creation Date To |

| | |
|---|---|
| Remove Person Data | Obfuscates the user name and display name of a user created in Oracle Identity Cloud Service (IDCS) as well as a person identified when ingesting business entity or case record data.<br><br>Enter the following parameters before submitting the program to run:<br><br>• Person Identifier - Enter the user name.<br><br>• Run Mode - Enter 1 or 2.<br><br>  • 1: Obfuscates the person data and creates a report which lists each impacted table and the number of rows affected. An example of obfuscation is replacing a user JSMITH with a meaningless string such as RVMGILKR.<br><br>  **Additional Information:** An obfuscated user appears as Unknown User in fields such as Uploaded By and Submitted By.<br><br>  • 2: Creates a report of the person data to obfuscate in each table, which lists each impacted table and the number of rows affected..<br><br>• Attribute Names - Optionally, enter specific table names known to contain person data. The Remove Person Data process only searches these tables for the person data. This reduces processing time by reducing the number of tables searched. Leave this field blank to search all tables for person data.<br><br>**Important:** This background process removes the person data from AIAMFG. You must remove the user from IDCS separately. See: Removing Users from |

| | |
|---|---|
| | Oracle Applications [https://docs.oracle.com/en/cloud/paas/identity-cloud/uaids/remove-users-oracle-applications.html]. |
| Summarize Sensor Data for Discrete Manufacturing | Extracts the time series features for running predictions model and summarizes sensor data for discrete manufacturing organizations.<br><br>Enter the following parameter before submitting the program to run: Organization Code. |
| Summarize Sensor Data for Process Manufacturing | Extracts the time series features for running predictions model and summarizes sensor data for process manufacturing organizations.<br><br>Enter the following parameter before submitting the program to run: Organization Code. |
| Update Actuals for Prediction Model Targets | Extracts the actual values of the targets of the completed work orders for use in the confusion matrix. |

# Glossary

**5Ms**

The five facets of a factory: Manpower, Machine, Management, Material, and Method.

**AIAMFG**

Oracle Adaptive Intelligent Apps for ManuFacturinG.

**algorithm**

An algorithm is a set of steps to accomplish a task. AIAMFG uses algorithms to process information in a dataset. The algorithm you choose for a model processes information from the dataset (the input source) and compares the results (the output) to a target measure.

**API**

An Application Programming Interface enables one application to communicate with another application. There are different types of APIs, but AIAMFG uses REST APIs, a type of web service API, which enables two machines to communicate with each other over a network.

**Apriori**

An algorithm you can select when creating an Insight model.

**artificial intelligence**

A term used to describe when computers mimic cognitive functions, such as learning and problem solving.

**attribute**

An attribute describes a data object. For example, a work order (the data object) has a completion date (the attribute). The rows in a database correspond to data objects and the columns correspond to attributes.

**bin**

An attribute value range. Model target measure results are placed in bins. For example, if a model predicts a quality target measure, such as pH, the bins might be:

- Below Lower Limit: 0-7
- On Target: 7-8
- Above Upper Limit: 8-14

Each bin represents a classification range. Each target measure result is placed into a classification range (bin).

### boxplot

A boxplot graph has a box from the first quartile to the third quartile of data. The vertical line through the box represents the median. The whiskers stretch from the first and third quartile to the minimum and maximum.

### business entity data

Business entities represent individual instances of data objects, such as customer 1376 (the instance or business entity) for the customer data object. In a Customer database table, each row contains the business entity data for each customer.

### case record data

Case record data CSV files capture completed and in-progress work order data in a flattened file format. Each row in the spreadsheet corresponds to one work order and contains all related entity data such as item, operation, routing, and so on, in the same row.

### case record identifier

Similar to a key field in a database. In a case record CSV file, the case record identifier, or key field, is a either a work order number or a work order/serial number combination.

### Chi-Square

An algorithm you can select when creating a feature significance model.

### classification

See: bin.

### cloud service

Any software service made available through the internet.

### configuration

In AIAMFG, configuration refers to the process of setting up custom data, such as key performance indicators and time series feature sets.

### confusion matrix

A set of charts that display total predictions generated for a dataset model and the percentage of true and false predictions by comparing predictions with actual results.

### context information

In AIAMFG, context information is ERP data, such as the work order, operation, equipment instance, and so on, added to sensor data.

### contextualization

The act of adding context information to sensor data.

### CSV file

Comma-Separated Values file. CSV files can be opened by any spreadsheet program. Each row is a data record and the columns represent data fields.

### data ingestion

To upload structured enterprise and semistructured sensor data into AIAMFG using comma separated values (CSV) file templates.

### data lake

A location in the cloud that stores consolidated data.

### data mining

The process of discovering patterns in large datasets.

### data preparation

The process of creating a dataset that meets certain parameters.

### dataset

A set of data that meets certain parameters, such as data related to a particular item and for a certain time period.

### DCS (Distributed Control Systems)

A machine data acquisition system used to collect machine sensor data and provide the data in a CSV file format.

### Decision Tree

An algorithm you can select when creating an insights model.

### descriptive stats

The descriptive statistics for a model feature. For example, you can view descriptive statistics such as distinct count, mean, mode, and so on for a quality test result feature

in a model.

### entity

See also: business entity data.

### ERP (Enterprise Resource Planning)

A type of software that organizations use to manage day-to-day business activities such as accounting, procurement, project management, risk management and compliance, and supply chain operations.

### event

An occurrence that appears in the timeline viewer in relation to an entity. Includes Enterprise Resource Planning (ERP) events related to work orders, purchase orders, sales orders and other logistical movements of entities, and events captured by sensor devices.

### exception

Something different than planned, such as a change in the planned work order completion date.

### feature

A data attribute that you select to use as an input feature in a model.

### flattened record

A denormalized record. The flattened record contains data attributes that were once stored in multiple tables. In AIAMFG, each record contains one work order and its related entities, such as item, operation, routing, and so on.

### genealogy

The relationships between database objects. For example, for a particular product, view the related equipment used, sales orders, and other details.

### histogram

A graph where each bar groups numbers into ranges. Taller bars show that more data falls in that range. A histogram is an approximate representation of the distribution of numerical data.

### identity domain

Alogical namespace for users and groups, typically representing a discrete set of users and groups in the physical data store. Identity domains are used to identify the users associated with particular partitions.

### insights

Insights from historical data analysis enable business users to discover the hidden patterns between influencing factors and production metrics such as production quality and yield

### instance (equipment or resource)

A particular piece of equipment (or resource) when multiple pieces of the same equipment exists. For example, a filling machine, when multiple filling machines that are all exactly the same exist within the same plant.

### instance (software application)

A bare metal or virtual machine (VM) compute host. The image used to launch the instance determines its operating system and other software. The shape specified during the launch process determines the number of CPUs and memory allocated to the instance.

### key performance indicator

A quantifiable measure used to evaluate the success of an insights or predictions model..

### Laboratory Information Management System (LIMS)

See also: LIMS

### LIMS (Laboratory Information Management System)

Software that enables you to manage samples, test results and associated data to improve lab productivity.

### machine learning

Computer algorithms that improve automatically through experience. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions.

See also: artificial intelligence.

### machine alert

A machine notification of an event such as downtime or the need for maintenance.

### machine event detection

Also known as machine event analysis. Use imported machine sensor device data to identify machine events, such as actual start time.

### MES (manufacturing execution system)

Computerized systems used in manufacturing to track and document the transformation of raw materials to finished goods. MES works in real time to enable the

control of multiple elements of the production process

**metadata**

Data that provides information about other data. For example, an attribute's metadata could include the attribute's unit of measure.

**metrics**

Production measurements. For example, quality and yield measurements.

**model**

A mathematical model performed on a computer. A model is the algorithms and equations used to capture the behavior of the system being modeled.

**module**

A part of a program. Programs are composed of one or more independently developed modules.

**network (supply chain network)**

Visual display of the relationships between an object such as a purchase order, lot, serial number, work order, or sales order and the associated entities.

**node**

An object in a network.

**operation**

A step in a manufacturing process where you perform work on, add value to, and consume department resources for an assembly.

**PAA (Piecewise Aggregate Approximation)**

An approximation of a time series. Divide the original time-series into M equally sized frames and then compute the mean values for each frame. The sequence assembled from the mean values is the PAA approximation of the original time-series.

**parametric data**

Sample data from a population that can be adequately modeled by a probability distribution that has a fixed set of parameters.

**prediction**

The bin into which the target measure for a particular work order is most likely to fall, according to a particular predictions model.

**prediction alert**

A warning received if a product's target measure prediction falls within a bin that has

alerts turned on.

### predictor

An input variable or feature that influences a prediction.

### Random Forest

An algorithm you can select when creating a Feature Significance model.

### REST

REpresentational State Transfer. REST is a standard architectural style used when creating a web service.

### REST API

Also known as a RESTful API. A REST API is a type of REST web service. Web services enable two machines to communicate over a network.

### REST web service

A REST web service consists of the following components:

- URL Path: https://api.xxxxxx.com/aimfgapi/vof/v1/fsm/wipexception
- HTTP Method (GET, PUT, POST, and so on)
- Header (optional)
- Parameters (optional)
- Body (optional)

### roles

Persona assigned to an application user. Each role provides access to certain application modules, features and data relevant to their role.

### rules (algorithm)

The definitions for a sequence of operations in an algorithm

### SAX (Symbolic Aggregate approXimation)

A symbolic representation for complex time series data.

### SCADA ( Supervisory Control and Data Acquisition)

Acomputer system for gathering and analyzing real time data. SCADA systems are used to monitor and control a plant or equipment.

**seeded data**

An initial set of data provided with a database when it is installed.

**semi-structured data**

A form of structured data that does not obey the formal structure of data models associated with relational databases or other forms of data tables, but nonetheless contains tags or other markers to separate semantic elements and enforce hierarchies of records and fields within the data. Examples of semi-structured data in AIAMFG are stream and alert data.

**serial analysis**

Models predict target measure results for serial units rather than for work orders.

**serialized start operation**

The operation where the serial number is applied to the item.

**stratified sampling**

Samples drawn from a number of separate strata of the population, rather than at random from the whole population, in order to provide a representative sampling.

**stream data**

Data that is continuously generated by different sources.

**structured data**

Organized elements of data that related to one another provided in a standardized format, such as a relational database.

**SVM (Support Vector Machine)**

An algorithm you can select when creating a Predictions model.

**target measure**

A production metric.

**test data**

Data used to assess the performance of a model. A model's predictions are based on test data.

**time segment**

A period of time. For example, 5 minutes.

**time series data**

An ordered data set with data points in specified intervals.

**time series feature set**

Used for model building and analysis. Constructed from time series data divided into time segments and either simple (average, min, max) or advanced functions (SAX).

**timeline**

Displays a measured value as it changes over time.

**training data**

Data used to fit the model. After receiving the model result based on the training data, adjust the model's parameters to best achieve the target measure.

**unstructured data**

Data that is not organized in a pre-defined manner or does not have a pre-defined data model.

**web service**

a service offered by an electronic device to another electronic device, communicating with each other over the internet.

**work order**

a product, service, task or a job that can be scheduled or assigned to someone.

**Z-normalization**

A calculation that measures the number of standard deviations by which the value of a data point is above or below the mean value of what is being observed or measured. Used with SAX.

# Index

## G



## H



## I



## K



## L



## M

## N



## O



## P



## R



## S



## T

## U



## W