

# Engine Angel OnBoarding Processes

ENGINE ANGEL No Account? [Request one](#) [Sign In](#)

WHAT IS ENGINE ANGEL? HOW DOES IT WORK? BENEFITS WHAT IS REQUIRED? HOW DO I GET STARTED? CONTACT US

**Diesel, Gasoline, CNG, Bio-fuels**

**4-stroke AND 2-stroke**

*Works for all types of reciprocating engines*

ENGINE ANGEL No Account? [Request one](#) [Sign In](#)

WHAT IS ENGINE ANGEL? HOW DOES IT WORK? BENEFITS WHAT IS REQUIRED? HOW DO I GET STARTED? CONTACT US

**Hauling: People, Products, Resources**

**On Highway, Off-road, Water, On Rails, In the Air, Stationary**

*For most vehicles in the transportation industry*

ENGINE ANGEL No Account? [Request one](#) [Sign In](#)

WHAT IS ENGINE ANGEL? HOW DOES IT WORK? BENEFITS WHAT IS REQUIRED? HOW DO I GET STARTED? CONTACT US

*Go from data about the past to prediction of what is likely next and recommendations to make that future better!*

**ENGINE ANGEL IN ACTION**

- Fleet Management
- Large Engine Repair & Rebuild
- Used Vehicle Dealer

**KNOWLEDGEBASE**

- Background Information
- Case Studies
- Science & Engineering

**FREQUENTLY ASKED QUESTIONS**

- What is Engine Angel?
- How much does it cost?
- How do I sign up?

This page intentionally blank.

# Engine Angel Documentation

## Volume A. Customer OnBoarding

2020-04-26

### Contents

Volume A. Customer OnBoarding .....	3
Introduction .....	6
EA Organization – On-boarding .....	8
Engine Angel Standards .....	9
Authorization .....	9
Locale-specific data presentation .....	9
Time Intervals.....	10
Multiple Language Support.....	10
Engine Angel Data Layers (Flows) .....	12
<b>OnBoarding</b> .....	12
Vehicles - Background.....	14
Vehicle Models.....	14
Vehicle Types .....	14
Vehicle Vocations.....	15
Vehicle Chassis.....	15
Trailer Data .....	16
Trailer Types.....	16
Trailer Chassis (use Vehicle Chassis screen).....	17
Engines.....	17
Engine Occupations.....	18
Engine Models Maintenance .....	19
Expense Items- Background.....	20
Axle Types .....	20

Tire Brands .....	21
Tire Lines .....	21
DOT Tire Codes.....	21
Brake Brands .....	22
Battery Brands .....	22
Inventory Expense Groups .....	22
Code Lists .....	<b>Error! Bookmark not defined.</b>
Security Data .....	23
All People .....	24
Fleets.....	24
Account Requests.....	24
Localization Data.....	24
Languages.....	25
Postal Codes.....	25
Global Facilities .....	25
UoM Rule .....	26
UoM Data.....	26
Alerts/Content .....	27
CMS Editor .....	27
Email Templates.....	27
Alert Criterion .....	28
Alert Definitions .....	29
Work Request (Templates) .....	30
Work Request – Data <sup>1</sup> .....	31
Wizards Data – Processes .....	32
Subscription Types .....	32
Remote Data .....	32
Data Connects.....	32

Translation ..... 33  
Translate CodeLists ..... 33

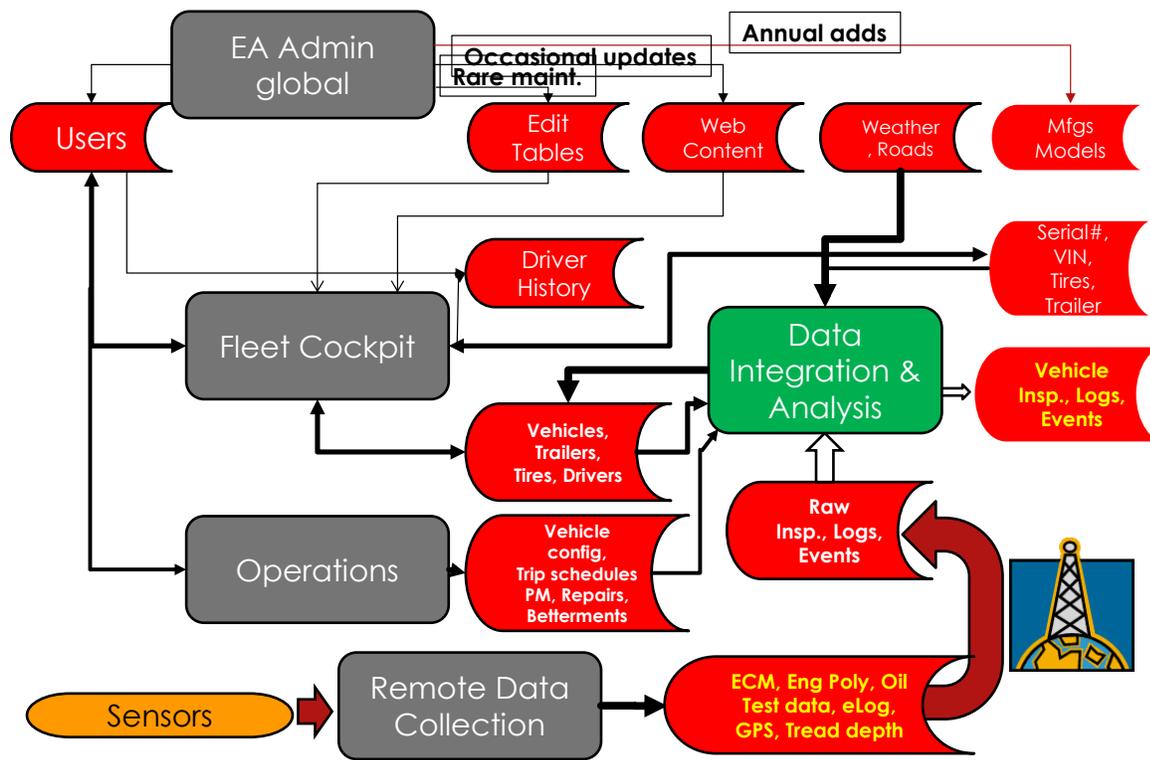
# This document is under construction

# Introduction

The documentation for Engine Angel is provided in 5 volumes of system functions and data organized by the sources of data and information – people, sensors, and analytical processes.

1. **Admin Layer (This document)** The **Admin** Layer for the staff of Predictive Fleet Technologies who are responsible for much data that is independent of any customer Fleets (equipment specifications, conversion standards, regulations, industry standards, Expert System rules, etc.)
2. The Fleet **Cockpit** for managers, professional staff responsible for the policy and management of operational people to achieve the organization's goals (managers, analysts, clerical people)
3. The **Operations** people who implement the actions according to policies and guidelines of management resulting in organizational success.
4. The **Remote** functions are those processes that connect sensors and special purpose accumulators of 'short-term' histories of data and transmit, transform and integrate the data into the system's databases to keep data about asset status and activities accurate and timely.
5. The **Analytics** functions (data integration and analysis) are processes that monitor asset status data against preset threshold values to alert operations and staff personnel of abnormal situations that might need corrective action. In addition, the batch processes use the historical data and models to predict future statuses and compare against the preset thresholds to provide proactive alerts and, using Expert Systems, suggest corrective actions to avert problems.

The following diagram (flow chart) represents these functions and major flows of data through the system.



# EA Organization – On-boarding

EA is also supported through distributors. There is a significant on-boarding process to collect information on your assets, employees, facilities. In addition, the history of maintenance, repairs and inspections and asset ECM histories are needed to initialize the models. Assets are assigned a vocation while both assets and facilities are assigned a home base.

Employees are assigned roles that activate permissions for data entry & correction by business process. The overall concept is that most data is visible to all employees. Roles are organized into 'levels':

**Cockpit** for managers and supervisors to view performance and control things like alert triggers, workflow sequences, work order/inspection formats; and an

**Operations** layer for drivers, technicians, procurement, dispatch, etc. who operate the assets to generate revenue.

**Remote** layer in the system controls the interfaces to data sources like ECM trip data extractors, EP from 3<sup>rd</sup> party shops, *engine oil test results, exhaust gas analyzers, work order, procurement, inventory management laptop applications.*

**Analytics** layer combines the data above to generate 'personalized' models of deterioration for many asset combinations, project future activity for vehicles, trailers, engines and drivers. These are combined with data showing current state of assets to *predict End-of-Life (EoL) dates* for generation of work orders, Alerts, asset ranking, etc.

# Engine Angel Standards

## Authorization

All authorized UserNames are assigned to either the Admin Layer (Predictive Fleet Technologies employees) or the Cockpit/Operations Layers (Customer employees). Once authorized to access a Layer, the UserName is allowed to view all data, except personal data of other people and data under the Security tab.

Authorization to update data is assigned to a UserName by assigning the UserName the corresponding Role (Authorization) on the 'All People' table in the Admin Layer for employees of PFT, or the Role using the Cockpit > Master Data > People table for the Cockpit and Operations Layers.

One UserName can access only the Admin **or** the Cockpit & Operations layers of EngineAngel, not both.

## Locale-specific data presentation

Event **DateTimes** are to be displayed in Fleet TimeZone; system DateTime is in UTC (as shown on the screens near the top right. Format is yyyy-mm-dd hh:mm:ss (I.e., 2017-01-25 22:07.843). In the csv files, the Date and Time are separated by an '\_'.

Date input fields use a Calendar pull-down in a popup window for ease and consistency of input.

**Units of Measurement** are to be metric in the Admin layer and as specified in the Fleet configuration data of Cockpit > Manage Fleet(s) > IsMetric (Y/N). Specific data elements can be defined in the Admin > Localization > UoM Data Rules. Conversions between various UoM are to be calculated using standard routines (SP) from data stored in the UoM Data table.

Distance calculations from GPS coordinated are to use the **Haversine method**.

### **GPS coordinates are to be stored in the DD (Decimal Degrees) format**

Longitude values West of the Greenwich median are stored as negative numbers and positive for East of the meridian. Values must be between -180 and +180 so that if less than -180, add 360; if over 180, subtract 360.

Latitude values are between 0 (equator) and 90 at the North pole;  
-90 is the South pole. If a latitude is >90, subtract 90; ; if less than -90, add 90.

Altitude is to be in meters (m).

There are three formats on the sidewalls of tires (**Tire size**):

wwldd.d → ww is the width in inches. Convert to metric (cm) for database.  
l=internal construction=[R,D,B], usually R=Radial

dd.d = diameter of wheel, normally to .5". Convert to cm for database.  
Assume aspect ratio = .9 so section height = .9\*ww

www/aaldd.d → www is width in mm. Convert to cm by dividing by 10;  
aa = aspect ratio → section height = (aa/100)\*width;  
dd.d and 'l' are like above.

www/aalddd → www is width in mm. Convert to cm by dividing by 10. Section height =  
(aa/100)\*width;  
ddd = diameter of wheel in mm, ,convert to cm for DB by divide by 10.

Overall diameter = wheel diameter + 2\*(Section height)

7.50R16, 245/75R16, 190/65R390 are examples of each

## Time Intervals

The human display for time intervals is of format: dd.hh:mm:ss (dd=days, hh = hours, mm = minutes, ss = seconds).

Time interval values are stored in db as seconds (s)

There are a number of components required for supporting multiple languages in Engine Angel.

## Multiple Language Support

The UTF-8 codepage is required in the various text fields to support the various character sets, and ideograms of the various languages of the world.

Language codes are the 2-character, lower case ISO codes.

Since the Admin Layer is only for PTF access, it is designed for users with English literals and messages. However, data is to be visible in any selected language. Adds, changes, and deletes of codes can only be made in English since that is the language code that will update relationships between codes. Translations are managed by separate, usually batch, functions.

A language must be specified for the fleet which will be the language used for all batch reports.

The user may identify a different language for display of data at the time of sign-on (with ability to change during the session). This will enable the system to display standard descriptions, field titles, and error messages in the language requested. Comments made by other members of the fleet should be made and will be displayed in the fleet language.

Field titles and literal texts provided by the GUI are maintained in a ML-Contents table in the various supported languages.

The CodeLists table identifies codes in their relationships in a set of tables maintained in the Admin layer under the classification menu. The translations of those codes, even into English, can be provided with the batch functions in the translations menu of the admin layer. They are stored in ML- extension tables. The CodeLists and Codes can only be added, deleted or changed only in English when using the online screens; non\_English Codes and Descriptions can be viewed in the Admin layer by changing the language (Locale) selection.

A number of tables contain language specific descriptions of groups and objects. These tables have ML- extensions for the corresponding translations. The maintenance of the translations are provided by requested downloads of the English content with the corresponding local language content; these tables are sent to translators, updated, and returned for upload into the ML- extension tables.

*Field edit messages in maintenance programs are extracted from the programs and stored in the 'messages' table. ML-extensions provide the local language equivalent after translation.*

Texts associated with Alerts, Work Orders, and other batch generated reports for users are maintained in the admin layer under the Alerts/Contents menu.

The following pages in this document detail specific features of the MultiLingual capabilities.

ADMIN => Localization => Languages => (below list screen)

EngineAngel > Translation Foundation > Enabling Engine Angel Translation.pptx

EngineAngel > Translation Foundation > ML Table Maintenance.pptx

EngineAngel > Translation Foundation > MultiLingual Application Messages.pptx

EngineAngel > Translation Foundation > CodeList Tasks.pptx

EngineAngel > Translation Foundation > Translation Tools for Translators.pptx

## Engine Angel Data Layers (Flows)

The Engine Angel Application functions can be grouped by Interactive Functions, Remote Functions (such as data collection and integration from sensors and special purpose computers) and scheduled (batch) functions to analyze data for status changes and predictions to generate Alerts to people to consider 'corrective' actions.

## OnBoarding

In preparation of the implementation of Engine Angel Fleet Management, there are a number of steps that must be planned to collect, harmonize, synchronize the data about your assets, employees, operations so we can establish baseline and trigger values for alerts and initialize the predictive models.

The configuration of your fleet is an important step.

- Setup of locations (sites) of your operations, Fleet language, units of measure, currency, etc. Fleet short name to be used in the system
- Deciding how to segment your fleet activities by vocation and location, if appropriate
- Establish model assets (skeletons that can be templates for addition of vehicles, trailers, engines, chassis structure)
- Verify ECM is loaded with correct vehicle identification, engine serial number
- Setup of the standard work order templates for inspections and Preventive Maintenance and Repairs by model assets
- Defining spare parts in the VMRS structure, standard cost, brands and models you use
- Defining 'Bins' in your facilities for parts inventory placement
- Collecting contact info for your employees (part-time and contractors included) and assigning roles they will have in the system
- Mapping your engine ECM details to pass required trip data to EA; setup of ECMDDataSync, telematics (including EDL)
- Identifying and authorizing outside services for engine oil analysis, Engine Polygraph signatures, exhaust gas analysis
- Defining how driver on-road expenses might be collected: credit card files for fleet cards? Manual summary inputs?
- Specification of additional electronic data sources, e.g., purchasing system, dispatch system
- Scan of existing work order and inspection systems to obtain:
  - Engine oil changes (and filter replacements)
  - Tire data: date of current tire placement, brand, last measured tread depth with date
  - Data from previous tires: vehicle/trailer it was on; date, odometer, tread depth at installation, date, odometer, tread depth at removal
  - Date of last DPF installation and cleaning
  - Date of installation of batteries, wheel bearing replacements, cabin filter,
  - Date of last Valve Lash adjustment
- Build baseline dashboard of
  - Oil changes on time

- Tires changed before tread-depth violation
- Brake replacements before pad thickness violations
- Number of DPF auto-regens and manual regens
- 
- Extract driver behaviors from ECM trip file with driver id. Cluster the profiles within vocation and location
- Establish start baseline for Driver retention, profile
- Generate weekly forecasts for vehicles, trailers, drivers
- Build ‘personalized’ models for tires, batteries, wheel bearings, DPF cleanings, oil changes
- 

Data from Admin

Automatic adds from ECM

Add-like Patterns

Templates with approvals

Implementation:

Hardware assessment: PCs, laptops, double screens, printers, internet access, WiFi router placement

Any translations required?

Strategy to handle the training: Overtime to learn new processes and methods? Temporary hires during startup? ‘Train the trainer’ first?

Implementation strategy: Priorities on urgent business drivers? Minimal impact on operations? Cold turkey? Management (Cockpit) first, Operations, External data sources?

## Vehicles - Background

Vehicles in the Admin Layer identify self-propelled units that typically carry or pull cargo and/or passengers that are available for Fleets to buy or lease and operate. At this level, the data is abstract, representing manufactures' model features and intended uses. The data is available for any Engine Angel customer to select from.

Fleets are provided Vehicles for on-road and off-road vocations. Marine Vessels are also Vehicles, but have not been loaded to the database tables as of the time of this document's publication. Stationary engines are installed in Facilities instead of Vehicles; a number of these engines are provided in our Engine tables, but we are far from complete and expect our customers will let us know the manufacturers and models relevant to current business needs.

### Vehicle Models

Vehicle Models are listed 'under' manufactures of models of vehicles for sale. The name of the manufacturer, model name, an acronym and an URL are available data elements. The acronym and URL are optional.

The primary purpose of the Model information is to provide data about the vehicles produced under the model name to users in the cockpit layer for easier and more accurate selection. The data that is input to the system comes from manufacturer spec sheets or other public sources.

The Vehicle data is maintained by PFT so it is possible that you have vehicle models in your operation that have not yet been added to the EA system. In such a case, please inform [support@PredictiveFleetTechnologies.com](mailto:support@PredictiveFleetTechnologies.com) about your need.

The Model record has a 'Default Chassis' configuration and a 'Default Drag Coefficient' that are presented to the authorized Fleet Master Data person to be confirmed or edited as appropriate.

A Model does not have a beginning or end date in the system. This means that a Model may be in production for some time period, but the produced model may be in operation for any number of years later.

Models are identified by common trade name used in the literature, often by the manufacturer's advertisements.

A Model may have a default Chassis attached and a default Aerodynamic drag coefficient.

### Vehicle Types

Vehicle Types are illustrated by an image that represents the shape and to a large extent, purpose of the vehicle.

The primary purpose of the Vehicle Type is to provide a technique for the Fleet personnel to assign the Vehicles they deploy into the fleet operation to a class so that the vehicles can be compared across a Type in the Fleet. The Type also provides information about additional data element requirements and options.

A Vehicle Type does not have a beginning or end date in the system.

Vehicle Types are identified by the stylized image but the shapes will vary over time and manufacturer.

The Vehicle data is maintained by PFT so it is possible that you have vehicle **models** in your operation that have not yet been added to the EA system. In such a case, please inform [support@PredictiveFleetTechnologies.com](mailto:support@PredictiveFleetTechnologies.com) about your need.

## Vehicle Vocations

Vehicle Vocations describe the industry that the Vehicle is expected to perform in. It also has an associated Application code to identify the environment that the Vehicle will operate in, e.g., urban, off-road, line haul, regional, etc.

The primary purpose of the Vehicle Vocation is to provide a technique for the Fleet personnel to assign the Vehicles they deploy into the fleet operation to a class so that the vehicles can be compared across a Vocations in the Fleet. E.G., Vehicles in a single Vocation are compared and ranked among themselves.

A Vehicle Vocation does not have a beginning or end date in the system; however, the Vocation of a Vehicle usually does go through multiple Vocations during its useful life; but Vehicle can be assigned only one Vocation at a time.

The Vehicle data is maintained by PFT so it is possible that you have vehicle **models** in your operation that have not yet been added to the EA system. In such a case, please inform [support@PredictiveFleetTechnologies.com](mailto:support@PredictiveFleetTechnologies.com) about your need.

## Vehicle Chassis

The Chassis record describes the basic chassis features of a Vehicle. This screen defines Chassis for both Vehicles and Trailers. It specifies the following features that are not likely to change over the life of the Vehicle.

The Chassis code has meaning to aid the user in selecting the appropriate code from a pull-down list: The first position is a 'L' if the Vehicle is a left-side drive (driver sits on the left side of the vehicle while driving) or 'R' for right-side drive; the second part of the code indicates the type of axle of the first axle of the vehicle (Axle Types are ['S' = steering; 'D' = drive; 'T' = trailer]) and followed by the number of consecutive axles that are of the same type. If any axle is a lift axle, it is coded by a preceding 'L'. So 'L-S1-LT1-D2' refers to a Vehicle that is left-hand drive, one steering axle in front, followed by a lift trailer axle, followed by 2 drive axles.

The Chassis record also specifies the number of batteries on the Vehicle/Trailer.

The Chassis code prompts the Fleet user to configure the wheel rims and tires, brakes on each axle and the battery set(s). The configured data defines the Vehicle/Trailer 'map' to prompt shop personnel for inputs during inspections and maintenance operations.

The number of Rims can be changed. But if that is done, a window will pop-up saying 'the number of tires changes when the number of rims change'. You must respond 'OK' to that message and then requested change must be Saved.

A new Vehicle/Trailer can be assigned a Chassis from the pull-down list, take the default from the prompt assigned to the Vehicle model or **copied from an existing, configured Vehicle**. The 'Distance on tire' will be inherited from the Vehicle/Trailer Master record.

The Vehicle data is maintained by PFT so it is possible that you have vehicle **models** in your operation that have not yet been added to the EA system. In such a case, please inform [support@PredictiveFleetTechnologies.com](mailto:support@PredictiveFleetTechnologies.com) about your need.

## Trailer Data

Trailers data in the Admin Layer identify non self-propelled units that typically carry cargo and/or passengers that are available for Fleets to buy or lease and operate. At this level, the data is abstract, representing Trailer Types and intended uses. The data is available for any Engine Angel customer to select from.

The Trailer industry is much more specialized around the world than Vehicles so that we do not provide Manufacturer nor Model data to the Fleets, but they provide that data in the Fleet Trailer record in the Cockpit Layer.

Trailers may have Engines attached at the Fleet level, e.g., reefers or cement mixers

### Trailer Data- Purpose

The purpose of the Trailer data is to reduce cost and time for data loads of customers and to provide consistency in the coding to support on-going customer evolution to increased system functionality and comparison of Trailers (performance, reliability, etc.) between Types.

## Trailer Types

Trailer Types are illustrated by an image that represents the shape and to a large extent, purpose of the Trailer.

The primary purpose of the Trailer Type is to provide a technique for the Fleet personnel to assign the Trailers they deploy into the fleet operation to a class so that the Trailers can be compared across a Type in the Fleet. The Type also provides information about additional data element requirements and options.

Trailer Type data can ONLY be changed and added in English. Translations of the description and code must be done with batch spreadsheets. Chassis and Drag coefficient cannot be different by language.

A Trailer Type does not have a beginning or end date in the system.

Trailer Types are identified by the stylized image but the shapes will vary over time and manufacturer.

A new Trailer Type can be 'copied' as a New Entry (Type) by checking the 'Save as New Entry' button and changing the data and image that is different.

A Trailer Type cannot be deleted unless it has no Trailer attached to it.

## Trailer Chassis (use Vehicle Chassis screen)

The Chassis record describes the basic chassis features of a Trailer. This screen defines Chassis for both Vehicles and Trailers. It specifies the following features that are not likely to change over the life of the Trailer.

The Chassis code has meaning to aid the user in selecting the appropriate code from a pull-down list: The first position is a 'L' if the Trailer is pulled by a left-side drive (driver sits on the left side of the vehicle while driving) or 'R' for right-side drive; the second part of the code indicates the type of axle of the first axle of the vehicle (Axle Types are ['S' = steering; 'D' = drive; 'T' = trailer]) and followed by the number of consecutive axles that are of the same type. If any axle is a lift axle, it is coded by a preceding 'L'. So 'L-LT1-T2' refers to a Trailer that follows a left-hand drive Vehicle, first axle is a lift trailer axle, followed by 2 Trailer axles.

The Chassis record also specifies the number of batteries on the Trailer, usually if the Trailer has an Engine.

The Chassis code prompts the Fleet user to configure the wheel rims and tires, brakes on each axle and the battery(ies). The configured data defines the Trailer 'map' to prompt shop personnel for inputs during inspections and maintenance operations.

A new Trailer can be assigned a Chassis from the pull-down list or copied from an existing, configured Trailer.

A Chassis cannot be deleted unless it has no Trailers attached to it.

## Engines

Engines in the Admin Layer identify manufacturers and their models for use by Engine Angel customers. This engine data is also available for use by the Engine Polygraph application to aid in interpretation of the FirstLook® Signatures.

There are other Engine/Engine Polygraph related tables under the Classifications menu option: Configurations, Conditions, and Channel Positions.

This data is collected and provided by Predictive Fleet Technologies to customers as an aid in the efficiency, accuracy and consistency of engine identification & specification.

### Engines - Purpose

The purpose of the Engine data is to reduce cost and time for data loads of customers and to provide consistency in the coding to support on-going customer evolution to increased system functionality and comparison of Engines (performance, reliability, etc.) between models. Some of the data is used by Engine Polygraph for the analysis of the EP Signatures and the 'Abnormal Observation' Diagnostics.

Much of the data is available via 'pull-down' lists in the Cockpit and Operations layers. Data content is made available where it supports implemented system function.

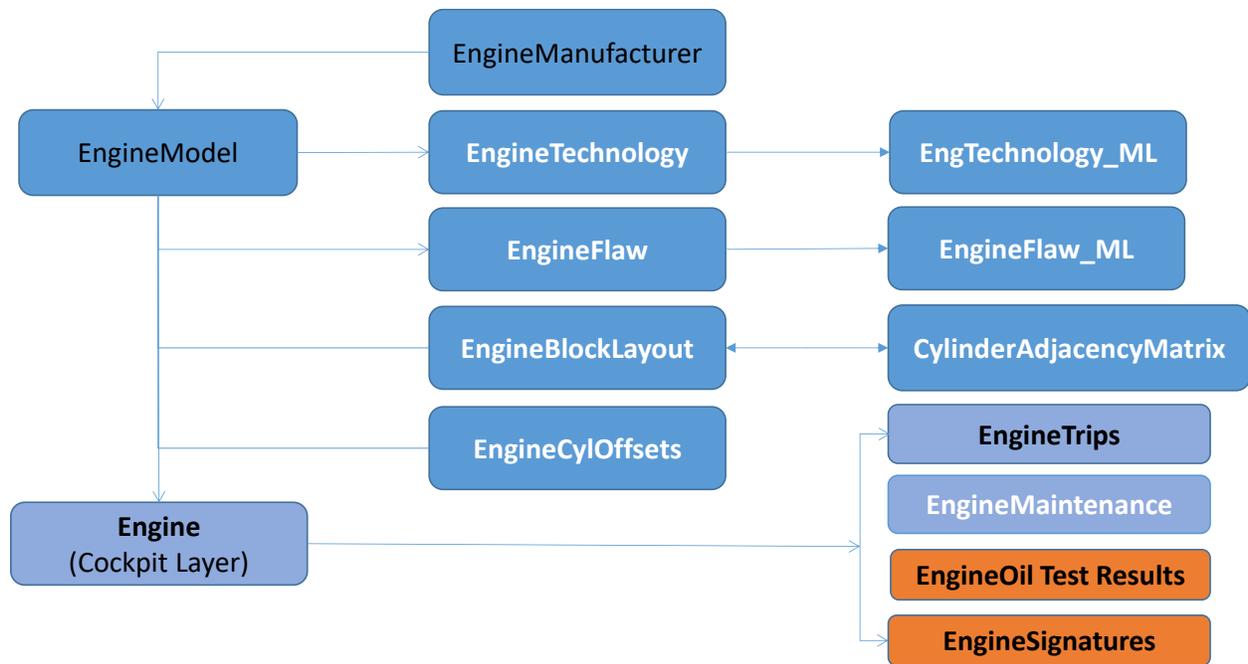


Figure xx. Relationships of a number of components of the Engine Model data in Admin and key data in the Cockpit layer.

## Engine Occupations

Engine Occupations describe the workload that the Engine is expected to perform under. The PM cycle is sometimes dependent on the Occupation that the Engine is in. E.G., Reefer and APU engines often run at constant rpm.

The primary purpose of the Engine Occupation is to provide a technique for the Fleet personnel to assign the Engines they deploy into the fleet operation to a class so that the Engines can be compared

across Occupations in the Fleet. E.G., Engines in a single Occupation are compared and ranked among themselves.

An Engine Occupation does not have a beginning or end date in the system. An Engine can be assigned only one Occupation at a time.

## Engine Models Maintenance

Engine Models are listed 'under' manufactures of models of Engines for sale. The name of the manufacturer, model name, an acronym and an URL are available data elements. The acronym and URL are optional.

The primary purpose of the Model information is to provide data about the Engines produced under the model name to users in the cockpit layer for easier and more accurate selection. The same table is used in the Engine Polygraph application. The data that is input to the system comes from manufacturer spec sheets or other public sources.

A Model does not have a beginning or end date in the system. This means that a Model may be in production for some time period, but the produced model may be in operation for any number of years later.

Models are identified by common trade name used in the literature, often in the manufacturer's advertisements.

**Engine Block Layout** identifier can be input in the Engine Block column based on the naming convention: Engine Configuration + Bank identifier that contains cylinder 1 [null, Right, Left] + cylinder numbering convention [Sequential, Alternating, Uturn]. The cylinder adjacency table is provided for each Block Layout. Images can be viewed before making an assignment by opening another tab to view the Block Layouts:

1. Click on the red bar with + in the upper right of the screen to get a new view of the Engine Models screen in another tab;
2. Click on the Engines menu option and click on the Engine Layouts option;
3. Scroll down the images to find the appropriate Block Layout and copy the 'Engine Layout' code;
4. Paste the value on the original Engine Maintenance tab, Engine Block column.
5. Then Save.

**Cylinder Offset Diagram** cannot be directly input but is built-up from other data elements: Number of Strokes, number of cylinders, and engine firing order. The image of the Cylinder Offsets can be observed from the **details icon**.

The **Technologies** button allows viewing previously assigned Technologies to this Engine Model or maintaining the relevant entries. Engine Technologies that are significant for interpretation of Engine

Polygraph waveforms for engine problem diagnosis. To add a new Technology to the list for this Engine Model, click on the 'Select technology' bar to get a list of unassigned technologies. You may enter a keyword to get a list of technologies that contain your input string; then, only technologies containing the string will be displayed. Click on the black + to the right of the selection list to add it to this Engine Model. Then click on **Save**.

The **Flaws** button allows viewing previously assigned engine model flaws or maintaining the relevant entries. Engine Flaws are commonly occurring problems with a specific Engine Model or set of Engine Models that occur with engines made between two 'dates' – from when engines with the flaw were first produced until the manufacturing process eliminated the flaw. To add a new Flaw to the list, click on the 'Select flaw' bar to get a list of unassigned flaws. You may enter a keyword to get a list of flaws that contain your input string; then, only Flaws containing the string will be displayed. Click on the black + to the right of the selection list to add it to this Engine Model. Then click on **Save**.

## Expense Items- Background

Expense Items in the Admin Layer identify manufacturers and their models for use by Engine Angel customers.

This data is collected and provided by Predictive Fleet Technologies to customers as an aid in the efficiency, accuracy and consistency of Expense Item identification & specification

The purpose of the Expense Item data is to reduce cost and time for data loads of customers and to provide consistency in the coding to support on-going customer evolution to increased system functionality and comparison of Expense Items (performance, reliability, etc.) between models.

Much of the data is available via 'pull-down' lists in the Cockpit and Operations layers. Data content is made available where it supports implemented system function.

## Axle Types

Axle Types describe the features for each axle type to aid in the Chassis configuration. Key features are:

- Description
- Number of axles in a group (e.g., tandem or tridem)
- A Position code [steer, drive, trailer]
- Image of an example.

All axles have bearings, brakes and rims. Any axle might be a 'lift' axle.

Steer axles require linkage to the steering mechanism to be checked periodically.

Drive axles have a differential that need lubrication and occasional replacement.

Trailer axles usually have distance measured with hubometers.

## Tire Brands

Tire Brands and manufactures are identified in this table. Tires are defined more by the Brand than the Manufacturer because the Brand provides the specification and might employ various manufacturers to produce tires under the Brand specification. The name of the manufacturer, abbreviation and Brand Name are stored with the URL for internet data. The acronym and URL are optional.

The primary purpose of the Tire Brands information is to provide data about the Tires produced under the Brand name to users in the cockpit layer for easier and more accurate selection. The Tire Lines (designs) are connected to the Brands and the Tire DOT Codes are connected to the Manufacturing plant to aid in recalls.

A Tire Brand does not have a beginning or end date in the system. This means that a Tire Brand may be in production for some time period, but the produced model may be in operation for some years later.

Tire Brands are identified by common trade name used in the literature, often by the manufacturer's advertisements.

## Tire Lines

Tire Lines are connected to the Brand that manages the Tire specification. The Tire Lines under the Brand define specific designs and sizes for various axle types.

A Tire Line does not have a beginning or end date in the system. A Tire Line can be assigned only one Brand at a time.

## DOT Tire Codes

Tire DOT Codes are provided to identify the Brand and location of manufacture of the tire to aid in case of recalls. The DOT Code and the week & year of production is branded on each tire sold in the US.

Format on Tire: 'DOT' indicates the tire meets DOT safety standards; 2-3 characters (DOT Tire Code) identifies mfg plant code; 2 digits = tire size; Brand code; Date of Mfg = week number of year and last 2 digits of year.

A new DOT (Plant) Code may be added directly or by using the change function on a previous DOT (Plant) Codes and clicking the 'Save as New Entry' button, making the desired change and then pressing Save.

## Brake Brands

Brake Brands are in the BrakeManufacturer table which really stores the Brake Brand names.

A Brake Brand does not have a beginning or end date in the system. A Brake Brand can be assigned only one Brand at a time.

## Battery Brands

Not yet implemented.

## Inventory Expense Groups

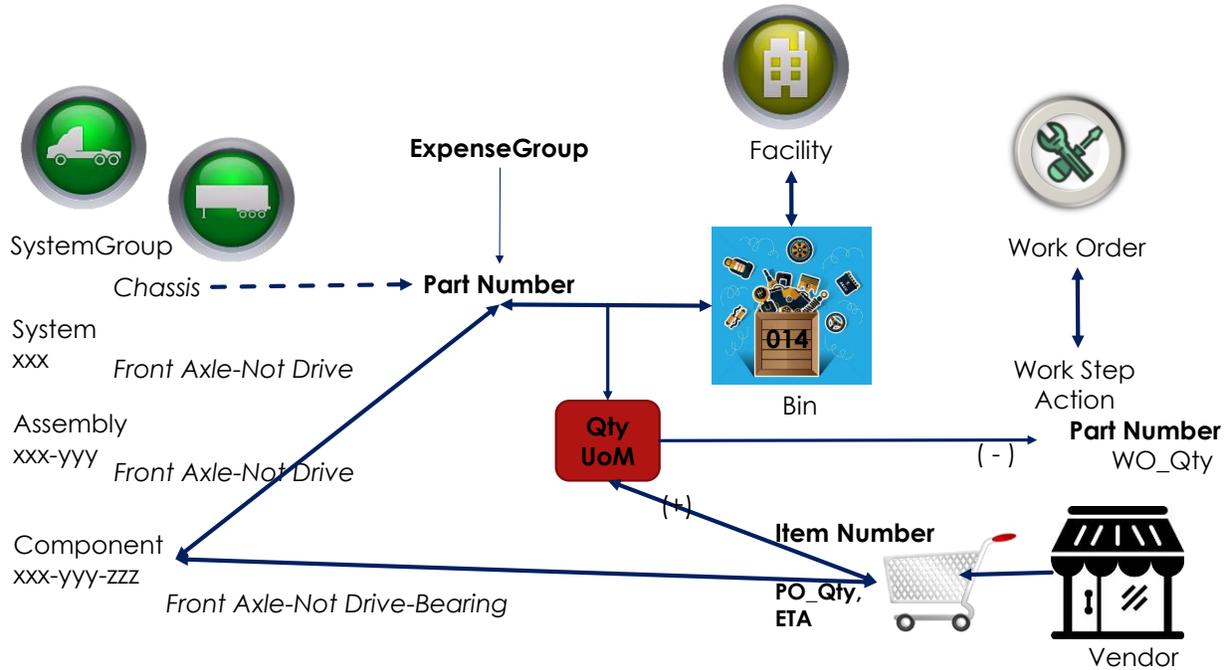
Not yet implemented.

Inventory Expense Groups (IEG) organize expense items into similar groups. For example, engine oils and hydraulic oils might be under lubricants.

IGEs relate to RepairSystems, a CodeList in the Classification system. Repair systems are components of the vehicle or trailer (unit) that together provide a specific feature to the unit. For example, the chassis includes the axles, bearings, suspension system, brakes, wheel rims and tires.

Expense items that are in inventory must be assigned to an Expense group and to at least one Repair System.

### **Relationships between Data Objects of Inventoried Items**



### Presentation of Codes to users:

- There should be common code for the retrieval of the values to present to users
- **Database records** should only use the entry from the CodeListDetail table
- **Displays and reports** should use the value retrieved from the User language (for screens), the Fleet language for Fleet reports, 'en' for reports where no language code is available, or as otherwise specified
- For **Pulldown lists**, get all the Detail (base) codes under the CodeListName, then find the equivalent for the appropriate language. The ML table has the id to the equivalent CodeListDetail so that plus language code provide quick access to the ML value if present. If not present, you already have the base (Detail) value. But store the base value in any database record.

## Security Data

## All People

The All People functionality allows authorized PFT employees to see and change 'People' data from any Fleet and to Add or Change information about Predictive Fleet Technologies employees, including authorization and status changes.

The primary purpose for the access of customers' employees is to enable Technical Support to customers who may have experienced critical employee turnover without adequate training of new employees.

## Fleets

The Fleets screen allows authorized PFT employees to view and change data about all Fleets in the Engine Angel system.

The primary purpose is to enable Technical Support to customers who may have experienced critical employee turnover without adequate training of new employees.

## Account Requests

Account Requests provide a listing and details for all un-processed requests from prospects to obtain a subscription to Engine Angel.

The options are to change and Save the changes about the request, Exit the screen without saving changes, or Send an invite to access the system.

To proceed to the invite, Fleet information must be provided to create the basic records in the Engine Angel system to enable the requestor to login.

## Localization Data

This Localization data is provided to enable standardization of data geographically and culturally as well as chronologically.

Where possible, we base our coding on ISO standards but also provide cross-walks to various industry and regional conventions.

Here we provide the options and in many cases, methods for conversions between standards. In a number of cases, the relationships change over time, e.g., Currencies and Currency Exchange rates.

The options are provided to the Fleet customers by means of pull-down lists.

- Countries
- State/Provinces
- Postal Codes
- Global Facilities*

Languages  
UoM Rule  
UoM Data  
Currencies  
TimeZones

## Languages

The Language table provides the list of languages that can be selected in Engine Angel and Engine Polygraph. That does not mean that the translations are complete in those languages, but that the system is prepared to accept the translations.

The Language code is the ISO two-character, lower case code that is connected to a 'Display Language' value that is the ISO code followed by the name of the language expressed in that language and to the 'Locale code' that combines the language code with a ISO Country code (3-character, capitalized letters).

The multilanguage capability is supported by several constructs:

The screen literals are stored in a Content table and associated Content\_ML for language-specific renditions for the term.

The system messages to the User in the GUI are stored in a GUIMessages table and associated GUIMessages\_ML table.

Codes and Code Descriptions for pull-down lists are stored in the CodeListMaster & CodeListDetail tables with the associated \_ML tables.

Other tables with language-specific fields that are to be made available in multiple languages are stored in the 'base' table and associated \_ML tables.

## Postal Codes

The Postal Codes are assumed to be managed at the national level. The Country code, Postal code, and State/Province code are provided to show the hierarchy.

The Postal Code Place name (usually City) is made available in a Pull-down list to increase consistency in address specification.

The TimeZone, GPS coordinates, and the Altitude are all stores as available defaults. A Postal Code status and type are maintained to indicate possible restriction on the use.

## Global Facilities

Global Facilities are made available with system function very much like the Cockpit > Master Data > Facilities management, except that here, the Facilities added from this screen are available to all Fleets (within a Fleet's assigned Country).

## UoM Rules

UoM Rules define exceptional Unit of Measure values in specific table columns. The Basic Rules are:

UoM for numeric values in the Admin Layer are metric,

UoM for numeric values in the Fleet Layers (Cockpit and Operations) are specified by the Fleet table (Cockpit > Account > Dashboard > Fleet Information).

Unless otherwise specified, the UoM is defined by the 'Base' UoM for the assigned dimension.

An Override Rule can be assigned to a specified column by selecting the table and then the column in that table. Values are to be provided for each of the 3 allowed UoM systems in the Fleets: [Metric, US, Marine]

**Unit of Measure Rules** define specific values for various types of data elements and in the case of traditional Units of Measure, conversion equations between units that are in the same Dimension and linearly related.:

**Numeric:** Numeric values can be unitless (e.g., counts) or dependent on a specified device to measure the quantity in a unit of a dimension; the relevant data are stored in the UoM Data table. Here we identify which UoM is to be used for the Fleet UoM System, as specified in the Fleet configuration.

If the Fleet specifies '**Marine**', distance and velocity will be stored in the unit identified in the 'marine' column.

If the Fleet specifies '**Metric**', the unit in the 'metric' column is to be used.

If the Fleet specifies '**US**', the unit in the 'US' column is to be used.

**TimeZone:** The TimeZones of the World and Fleet customization are stored elsewhere. Here we store whether the Fleet Language is to be stored or if UTC is to be used. (UTC is the standard for times in the Global tables and System columns (e.g., LastUpdateDateTime).)

**GPS coordinates:** Here we specify which of the two formats of GPS location is to be used: DD = Degree Decimal, or DMS = Degree-Minute-Seconds. A standard function is to be used to perform the conversions between them for consistency. DD is the format to be used in the 'Global' tables.

**Language:** The language codes and descriptions are stored in the Language table. Here we specify if the Fleet language or English is to be used.

**Currency:** *Details for conversion and options are stored in the Currency table. Since the conversion rates dynamic, rules for which rate to use must be specified [Current, Budget (time period), transaction date, etc.]. Here we identify 'Fleet' currency or USD.*

## UoM Data

The UoM Data table contains common acronyms and symbols used to specify units of measure. Many of these are set by ISO sub-groups, but others are conventions of use in various disciplines.

The table also identifies a 'Base' UoM for each Dimension; that Base unit is used to enable 'pivot' conversions from any UoM to any other UoM in the same Dimension.

Each 'metric' UoM is provided the typical equivalent UoM in the non-metric system, and vice-versa.

The coefficients of the linear equation that convert a quantity in one UoM to another UoM in the same dimension are provided. There is a standard function that employs these coefficients to perform the conversions.

The UoM maintenance screen has a Test feature to convert an input number from the Test Input Number of Unit Code to Calculated Base Units. The program also shows what would be the result if the input number were in the base units and calculates the equivalent number of selected code units and puts the result in Reverse calculation of input Base units to unit Code.

The System of Record is provided as a reference to check the accuracy and completeness of the data.

## Alerts/Content

### CMS Editor

The CMS Editor is a WordPress editor used to maintain the Home Page screen.

### Email Templates

Email templates are provided to standardize various communications from the system to Users and potential Users. The language specific texts are provided in multiple languages for several purposes:

- The Account Invite message welcomes a new prospect to the EngineAngel application for creating the Fleet in the system.
- The Reset Password request message is sent to the user who requested to reset his/her password.

Other common messages that are to be provided in multiple languages are to be maintained with this method.

## Alert Criterion

### Alerts/Content Data

As data is analyzed and events predicted in Engine Angel, data & predictions can be compared with triggers setup by the Fleet management for each Vocation to have the system alert them to situations that deserve their attention.

The alerts will be sent via email and logged into a Fleet table of Alerts sent.

Alerts will be classified as 'Important' or 'Urgent'. (Urgent are usually 'past due' items.)

Store Alert texts (Language specific) in Admin Layer under **WebSiteContent**

Incorporates variables from the event (embedded '{{0}}' by calling program)

### Alert Hierarchy

1. Criteria for Alerts **are grouped** into AlertCriterionGroups that are defined in a CodeList: AlertCriterionGroup
2. An Alert Criterion **is assigned** to an AlertCriterionGroup in Admin level > Alert Criteria
3. The Alert Criterion **is defined** in Admin level > Alert Definitions

SourceSystem (must be established on CodeList: AlertSourceSystem) identifies which function in EngineAngel has the code to test operational values against Criteria to trigger the Alert.

SourceKey (free text) identifies what variable is tested.

Description (free text) of the situation when the Alert is triggered.

Priority (on CodeList: Priorities) indicates the priority of the Alert in terms of which of multiple Alerts should be addressed first.

Format (CodeList: MsgFormat) indicates the content format of the message content of the template for this Alert.

Routing (Role) of people to receive the Alert in this Fleet) – assumes there may be more than one person in the Fleet's Role.

**Criteria** are 'configured' for the Fleet by two functions under the Fleet Maintenance menu item: Cockpit > Account > Account Dashboard > Manage Fleet (pencil) > Fleet Information > Fleet Alert Criteria.

Fleet Alert Criteria are set for Vehicles assigned to particular Vocation. (Extreme conditions vary for the equipment and driver behaviors depending on what the job is: Line haul, delivery, construction, passenger transportation, etc.)

The criteria available (that have been programmed) are obtained from the Admin level Criteria Group > Alert Criterion

The Values assigned by the Fleet are assumed to be in the Fleet's UoM system.

A specified 'Low' value means if the observed value is less than the indicated value, the Alert should be generated, If there is no 'Low' value in the Fleet Criterion for this Vocation, ignore this test.

A specified 'Hi' value means if the observed value is greater than the indicated value, the Alert should be generated, If there is no 'Hi' value in the Fleet Criterion for this Vocation, ignore this test.

## Alert Definitions

Alert Definitions are descriptions of the Alert situation that are sent to appropriate personnel in the Fleet based on assigned Roles. The list of PFT provided messages are listed with many of the codes describing the Criterion, Source system, Source Key, Priority and Title along with the Role of a person designated to receive (and resolve) the issue.

The add, Change or Delete can be requested for any Definition. Notice that the priority can be different for different values of 'trigger' variable. Usually, 'urgent' is reserved for issues that are past 'End of Life' (EoL).

The details screen allows specification of the Alert and the text of the Alert message to be presented. The messages can be text strings or HTML format. The content can include variables from the data associated with the Alert. Variables to be included must be surrounded by double brackets: '{{' and '}}'. Any translation of the Alert must employ the same variables but the syntax of the language will dictate the sequence.

## Alert Routing

Since texts are in batch, the messages will be in the language of the To: FleetRolePerson (Default to the Fleet Language) as contained in table: FleetAlertRouting

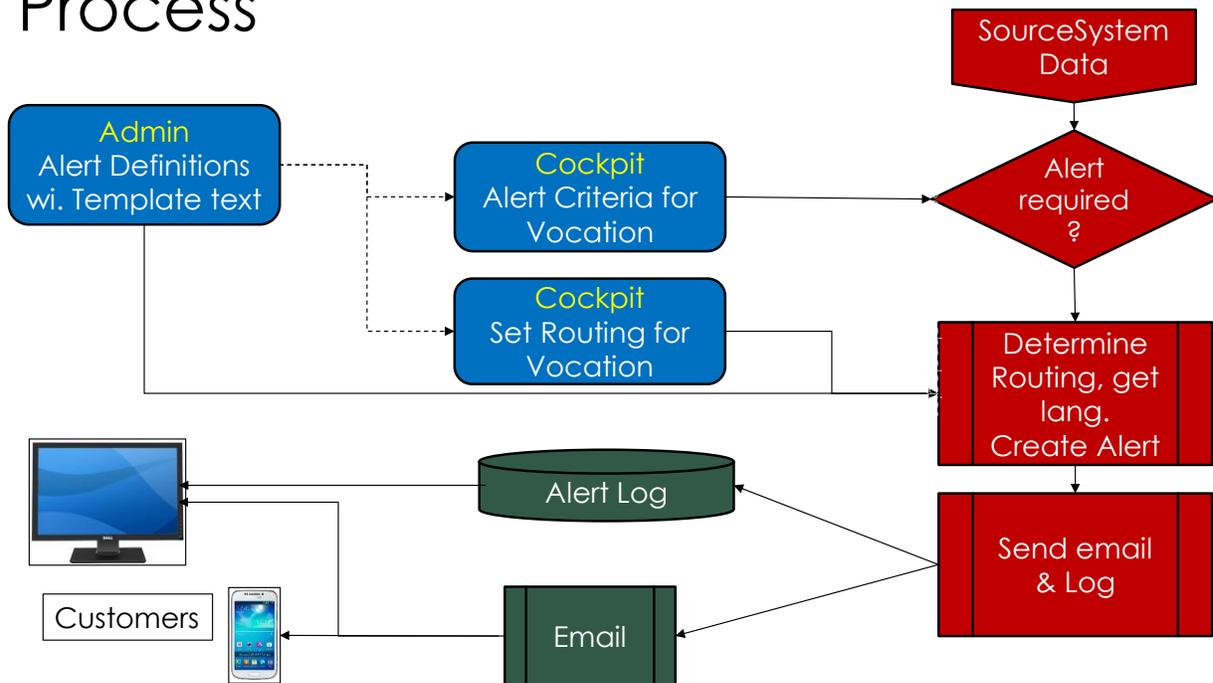
The Alert will be associated with a Fleet 'Role' default assignment to all people assigned 'Role' in Fleet

Alerts are responsibility of only one Role

Fleet can alter the distribution by specifically routing to UserName/email address by Vocation of Vehicle

It is important that system doesn't flood employees.

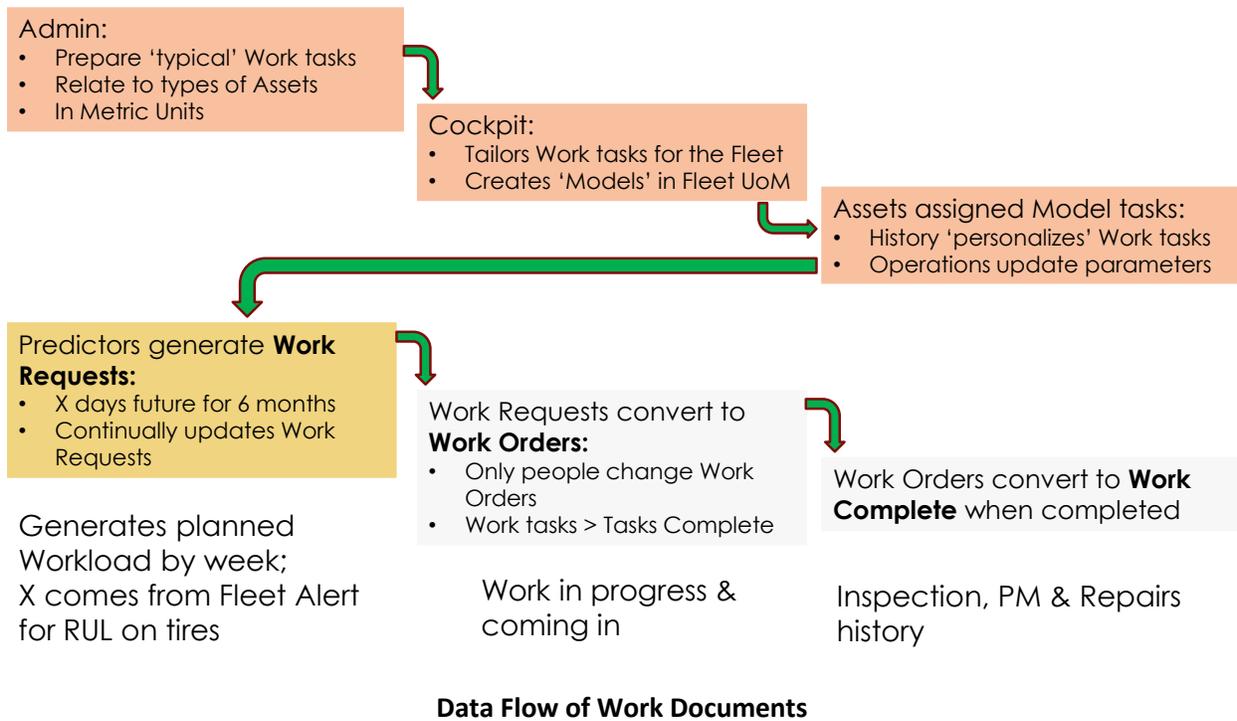
# Process



## Work Request (Templates)

Work Templates are defined at the Admin Layer to provide a starter set of Preventive Maintenance, Inspections, Work Order for repairs and betterments, etc. for any Fleet to maintain. Once a template is 'completed', it becomes visible to the Fleet on the Cockpit > Dashboard > Work Forms tab for download. Once downloaded, the Cockpit user may tailor the document to the standards and needs of the Fleet requirements. When a Work Form is 'completed', it becomes available in the Master Data > [Vehicle, Trailer, Driver] > Work Request Forms. Here, once the form has been approved for use by the technicians, it is marked 'completed'; it may then be 'assigned' to specific Assets [Vehicle, Trailer, Driver] or to reference vehicles with a leading 'REF: ' in the Vehicle Number.

The Assigned Work Request Forms are associated with 'wear rate' prediction rules, incorporating data from trips and inspections to predict the Remaining Useful Life (RUL) and End of Life (EoL) for the component base on forecasted daily usage of the Asset. The PM should be scheduled to be completed before that EoL Date.



## Work Request – Data

The data in a Work Request is in two types of tables:

The header provides:

Title – Descriptive identifier to aid in retrieval of the Work Order Form

Work Type – from a CodeList that defines the prediction logic for the specific Asset component wear-rate.

Schedule – Suggests the primary type of logic used to schedule the Tasks of the Work Request.

Status, Form No., and Version for document control.

Resource Type, Asset Type, and Vocation to provide context for the Asset usage

Typical intervals in terms of distance, hours of operation (HoO), or Calendar days (as for license renewals or government filings)

Typical incurred costs such as hours the Asset is Out of Service for the PM, Labor hours involved in the PM, and cost of Parts & Supplies for a typical WR.

The second type of table in a Work Request contains parameters specifying the data elements for the Tasks of the Work Request; note that it does NOT specify the contents of the Task lines.

The Detail Lines define each Task to be performed and documented using fields specific to the task:

Section & Seq (number) isolate the step into sections & tasks within the section

Title indicating the action to be taken

AxleMap, BrakeMap, TireMap indicators if the 'maps' are to be provided for the technician to record the data

CSA to show the CSA points involved with an inspection showing non-compliance with the stated claim for the task

Tech Condition Comment on the results of the task

Action

Measure

Part Number

Picture

Damage Cause

Part Unit Cost

Part Units

Labor hours, DownTime (in hours)

The marked indicators specify the data elements that are expected to be available for input on the final document.

## Wizards Data – Processes

### Subscription Types

### Remote Data

### Data Connects

DataConnects documents the established (supported) interfaces from 3<sup>rd</sup> parties to Fleets. The details of any interface between any Fleet and 3<sup>rd</sup> parties are stored at the Cockpit layer.

The list screen shows the set of eligible interfaces with some description and link to the software that provides the interface. The list of supported interfaces is controlled by table DataConnectType.

The current supported interfaces are:

1. Collecting, transmitting, and integrating ECM data from Detroit engines with DDEC software.
2. Accepting EnginePolygraph® (EP) signatures from PicoScope oscilloscopes into the web-based Engine Polygraph application.

3. EPReader: Here we provide the software for deployment on shop PCs to collect and transmit EP signatures to the EP Analytics and integration of the results into the EA Engine Condition reports.

#### Remote Data - Processes

In the future, we intend to include integration with additional OEM ECM softwares, *Oil Test Results*, and with *Telematics* (with Electronic Logs) .

## Translation

### Translate CodeLists

The Translations tab currently has one entry: Translate CodeLists.

The purpose of **Translate CodeLists** is to provide a tool enabling the use of XML spreadsheets to provide changes in CodeList content to translators and accept their new translations for updating the CodeList tables.

The Translate CodeLists screen contains two panels:

#### **1. PROVIDE DETAILS TO GENERATE EXCEL**

This screen provides an XML spreadsheet for Excel to show all codes in the CodeList system with an English description and a description for the indicated language and such that either has been modified 'recently' (within the last 'No. of Days' input on the screen).

#### **2. IMPORT TRANSLATED FILE**

The second panel allows the user to specify the name of an XML Excel table that contains updates to the Codes and Code Descriptions from the translators to update the table.

For a code to be considered for the download, it must have:

1. An English segment with the Code and Description in English
2. A segment for the specified language (perhaps filled with the base code and description or with an entry translated by Google or someone else)
3. The Last Update DateTime for the English or the specified language must be more recent than the current date minus the input 'No. of Days'.

The downloaded XML file is to be opened in Excel as an XML table. Changes to the table are to be made in the last three columns on the right of the rows: Date, Code, and Code Description. Date should be the Date that the update of the row is made and the rightmost Code and Code Description are to be changed to the appropriate translation.

If no changes to a row are made, that code will not be updated in the table when later uploaded.

The Excel file must be saved as "XML data" from the SaveAs option.