

MES AI User Guide

Release 1.0.0

Introduction	2
Installation.....	2
GitHub Repository	2
Database	3
PostgreSQL.....	3
MSSQL or Oracle	4
Database Schema.....	4
Python and Project Dependencies	4
Windows	4
Linux or macOS.....	5
MES Server	5
Windows Console.....	6
Windows Service.....	6
Linux or macOS Console	6
Linux or macOS Daemon.....	7
Client Applications	7
Windows.....	8
Linux or macOS	8
Client Startup.....	8
First Run	8
Demonstration Applications.....	9
Authentication	9
Local	9
OIDC OpenID Connect / Enterprise SSO	9
Settings Page Configuration	9
Design Time Client (dt-client)	10
Work Cells, Equipment & Equipment Classes.....	11
Routes and Dispositions	14
Routes	14
Dispositions	25
Products	26
Materials.....	28
Storage Locations.....	29
Units of Measure	30
Data Definitions	31
Reason Codes	31
Users and Roles	32
Work Schedule.....	34
Plugins	37
Enterprise Plugins.....	39
Factory Floor Plugins	41
Settings	42
Runtime Client (rt-client).....	42
Dashboard.....	42
Dispatch.....	43
Execution	44
Status Signals.....	45
WIP.....	45
Orders	49
Inventory	50
Equipment	52

Live Events.....	54
ERP Simulator (erp-sim).....	55
Inbound.....	55
Production Orders.....	55
Materials.....	56
Products.....	57
Outbound.....	58
Report Completion.....	58
Report Consumption.....	59
Report Scrap.....	59
Confirmations.....	60
Equipment Simulator (equipment-sim).....	60
Dashboard.....	60
Equipment.....	61
State History.....	62
OEE Analysis.....	63
Auto-Simulator.....	63
State Models.....	64
Software Quality Assurance.....	66
REST API Unit Tests.....	66
User Interface Tests.....	66
Documentation.....	66
REST API.....	66
Architecture.....	67
Further Study.....	67

Introduction

MES AI is an open-source Manufacturing Execution System platform built around a FastAPI server, multiple browser clients, and a plugin-oriented extension model. The repository includes installation and startup scripts for local development on Windows, Linux, and macOS.

This guide covers:

- cloning the GitHub repository
- installing and preparing the database
- installing Python and project dependencies with the provided install scripts
- starting the MES AI server
- starting the available client applications

Installation

GitHub Repository

Open a terminal in the directory where you want the project to live, then clone the repository.

```
git clone https://github.com/point85/mes-ai.git
cd mes-ai
```

If you prefer SSH and have GitHub SSH access configured:

```
git clone git@github.com:point85/mes-ai.git
cd mes-ai
```

Or, the zip archive can be downloaded from the GitHub web URL: [point85/mes-ai: Manufacturing Execution System powered by AI](https://github.com/point85/mes-ai).

Database

MES AI supports PostgreSQL, Microsoft SQL Server, and Oracle.

- PostgreSQL can be installed with the provided installation script.
- MSSQL and Oracle are supported by the main installer and server launcher, but the database itself must already exist before the main installer runs.

PostgreSQL

Use the PostgreSQL helper script:

`.\install-postgresql.ps1` on Windows or `.sh` on Linux.

After installation:

1. Start the PostgreSQL service if it is not already running.
2. Set the postgres user password (postgres is the default)
3. Run the main installer.

Example password commands:

Windows: `psql -U postgres -c "ALTER USER postgres PASSWORD 'your_password';"`

Linux: `sudo -u postgres psql -c "ALTER USER postgres PASSWORD 'your_password';"`

MacOS: `psql postgres -c "ALTER USER postgres PASSWORD 'your_password';"`

MSSQL or Oracle

If you are using SQL Server or Oracle instead of PostgreSQL:

1. Install and configure the database server separately.
2. Create the target database before running MES AI migrations.
3. Make sure the required database drivers are available.

Driver notes:

- MSSQL requires a usable SQL Server ODBC driver.
- Oracle requires the Python `oracledb` package, and may also need Oracle client libraries depending on your connection mode.

Database Schema

Database table names follow the ISA-S95 specification. There are no views or indexes created by the SQL Alchemy code. SQL Alchemy data types are used for portability. Alembic is the migration mechanism.

Python and Project Dependencies

The main install scripts create a local virtual environment, install Python dependencies, write `server/.env`, and run Alembic migrations.

Important behavior:

- `DbProvider` is required. There is no default database provider.
- `DbName` defaults to `mes_ai`.
- If you supply a full database URL, the installer uses it as-is.
- Automatic database creation is implemented only for PostgreSQL.
- If you do not supply any arguments to the install script, it will print `usage/help`

Windows

Use PowerShell:

```
.\install.ps1 -DbProvider PostgreSQL -DbPassword your_password
```

Examples:

```
.\install.ps1 -DbProvider PostgreSQL
.\install.ps1 -DbProvider PostgreSQL -DbName mes_ai -DbPassword your_password
.\install.ps1 -DbProvider MSSQL -DbServer sql-host:1433 -DbUser sa -DbPassword secret
.\install.ps1 -DbProvider Oracle -DatabaseUrl oracle+oracledb://mes:secret@oracle-
host:1521/mes_ai
```

Linux or macOS

Make the script executable, then run it:

```
chmod +x ./install.sh
./install.sh --db-provider PostgreSQL --db-password your_password
```

Examples:

```
./install.sh --db-provider PostgreSQL
./install.sh --db-provider PostgreSQL --db-name mes_ai --db-password your_password
./install.sh --db-provider MSSQL --db-server sql-host:1433 --db-user sa --db-password secret
./install.sh --db-provider Oracle --database-url oracle+oracledb://mes:secret@oracle-
host:1521/mes_ai
```

What the Installer Does

The main install script performs these high-level steps:

1. Verifies Python 3.12+.
2. Verifies Node.js 20+ unless client installation is skipped.
3. Creates the local .venv virtual environment.
4. Installs server dependencies.
5. Creates server/.env if it does not already exist.
6. Creates the database automatically only when using PostgreSQL without a full database URL override.
7. Runs Alembic migrations.
8. Installs npm dependencies for the browser clients unless skipped.

MES Server

The server startup scripts set the database connection string, apply migrations, and start the FastAPI application with Uvicorn. If not arguments are supplied, the script will print usage/help. There are two sets of scripts, one targeted for fast turnaround during development with console output and one for production as a Windows service or Unix

daemon. The development script is named run-server and the production one run-service.

Windows Console

Use:

```
.\run-server.ps1 <Database> [DbName] [options]
```

Examples:

```
.\run-server.ps1 PostgreSQL
```

```
.\run-server.ps1 PostgreSQL mes_ai -Username postgres -Password (ConvertTo-SecureString 'your_password' -AsPlainText -Force)
```

```
.\run-server.ps1 MSSQL mes_ai -Username sa -Password (ConvertTo-SecureString 'secret' -AsPlainText -Force)
```

```
.\run-server.ps1 Oracle mes_ai -DbServer oracle-host:1521 -Username mes -Password (ConvertTo-SecureString 'secret' -AsPlainText -Force)
```

Windows Service

The service name, description and arguments are set in the configuration file mes-service.conf. Edit this file for your specific service. To install the service and run database migrations, use an admin Powershell:

```
.\run-service.ps1 install -ConfigFile C:\dev\mes_ai\mes-service.conf -RunMigrations
```

The log files are in ~\server\logs folder.

Additional commands are:

```
Start now : .\run-service.ps1 start -ConfigFile "C:\dev\mes_ai\mes-service.conf"
```

```
Stop      : .\run-service.ps1 stop -ConfigFile "C:\dev\mes_ai\mes-service.conf"
```

```
Status    : .\run-service.ps1 status -ConfigFile "C:\dev\mes_ai\mes-service.conf"
```

```
Uninstall : .\run-service.ps1 uninstall -ConfigFile "C:\dev\mes_ai\mes-service.conf"
```

Linux or macOS Console

Use:

```
./run-server.sh <Database> [DbName] [options]
```

Examples:

```
./run-server.sh PostgreSQL
```

```
./run-server.sh PostgreSQL mes_ai -u postgres -p your_password
```

```
./run-server.sh MSSQL mes_ai -u sa -p secret
```

```
./run-server.sh Oracle mes_ai -s oracle-host:1521 -u mes -p secret
```

Supported database names for the launcher scripts:

- PostgreSQL
- MSSQL
- Oracle

Default server port:

- MES AI server runs on `http://localhost:8082` unless overridden.

Linux or macOS Daemon

The daemon name, description and arguments are set in the configuration file `mes-service.conf`. Edit this file for your specific service. First install the daemon and run database migrations. The commands are:

```
sudo ./run-service.sh install --config /etc/mes_ai/mes-service.conf --run-migrations
sudo ./run-service.sh start
sudo ./run-service.sh stop
sudo ./run-service.sh restart
./run-service.sh status
sudo ./run-service.sh uninstall
```

After installation, useful systemd commands are:

```
journalctl -u mes-ai -f      # follow logs (live)
systemctl status mes-ai    # detailed status
systemctl cat mes-ai       # view unit file
systemctl reload-or-restart mes-ai
```

The log files are in `~\server\logs` folder.

Client Applications

MES AI includes four browser-based clients:

- `dt-client` for the Design-Time client on default port 5173
- `rt-client` for the Run-Time client on default port 5176
- `erp-sim` for the ERP Simulator on default port 5174
- `equipment-sim` for the Equipment Simulator on default port 5175

These applications will run on Edge, Chrome, Safari and Firefox.

Windows

Use:

```
.\run-client.ps1 <Client> [options]
```

Examples:

```
.\run-client.ps1 dt-client  
.\run-client.ps1 rt-client  
.\run-client.ps1 erp-sim  
.\run-client.ps1 equipment-sim  
.\run-client.ps1 rt-client -ServerUrl http://localhost:8082  
.\run-client.ps1 equipment-sim -Port 5200
```

Linux or macOS

Use:

```
./run-client.sh <Client> [options]
```

Examples:

```
./run-client.sh dt-client  
./run-client.sh rt-client  
./run-client.sh erp-sim  
./run-client.sh equipment-sim  
./run-client.sh rt-client --server-url http://localhost:8082  
./run-client.sh equipment-sim --port 5200
```

Client Startup

- The client scripts install npm dependencies automatically if node_modules is missing.
- The clients are Vite development servers intended for local development and testing.
- The default backend URL used by the clients is http://localhost:8082.

First Run

For a typical PostgreSQL-based local setup:

1. Clone the repository.
2. Install PostgreSQL with install-postgresql.ps1 or install-postgresql.sh.
3. Run the main install script with DbProvider set to PostgreSQL.

4. Start the server with run-server.ps1 or run-server.sh.
5. Start dt-client and rt-client with run-client.ps1 or run-client.sh.

Demonstration Applications

To illustrate the capabilities of MES AI in two different industries, database seeding links have been implemented in dt-client:

- CPG Demo: creates an orange juice bottling line for a Consumer Packaged Goods industry. Juice is tracked by lots.
- Electronics Demo: creates a SMT (Surface Mount Technology) production line for electronic circuit boards for the discrete assembly industry. Boards are tracked by units.

Screenshots in this document have been captured from these apps.

Authentication

The authentication level (none, local or OIDC) is a setting in dt-client. "None" is the default and requires no credentials. The local and OIDC setting applies only to dt-client.

If changes are made and a user can't log in, run the script reset-auth.ps1/sh to set the authentication mode back to none.

Local

For local authentication, users log in with credentials stored in the MES database. Passwords are hashed with bcrypt; the server never stores plain text credentials. JWT (JSON Web Token) access token settings are also configured in dt-client.

OIDC OpenID Connect / Enterprise SSO

Authentication is delegated entirely to an external Identity Provider (IdP) such as Keycloak, Okta, Azure AD, or Auth0.

Settings Page Configuration

The relevant settings are grouped into two sections:

Authentication section

Setting	Description
Authentication Mode	Dropdown: none / local / oidc

JWT Secret Key	Masked; signs all JWTs — must be changed from default in production
JWT Algorithm	Dropdown: HS256, HS384, HS512, RS256
Access Token Expiry	Minutes until access token expires (default 15)
Refresh Token Expiry	Days until refresh token expires (default 7)

OIDC Settings section

Setting	Description
OIDC Issuer URL	Provider discovery URL (e.g. https://auth.company.com/realms/mes)
OIDC Client ID	Client ID registered with the IdP
OIDC Client Secret	Masked; write-only after first save
OIDC Scopes	Comma-separated scopes (default openid,profile,email)
OIDC Role Claim	JWT claim carrying the user's roles/groups (default groups)
OIDC Redirect URI	OAuth callback URL registered with the provider

Design Time Client (dt-client)

dt-client is the design-time application for configuring products, routes, dispositions, equipment, plugins, and other master data. The main editors are:

- ISA-S95 sites hierarchy to define organize equipment and define equipment properties
- equipment class to define properties of a type of equipment to be used in dispatching WIP to a piece of equipment
- route editor to configure product routing.

Supporting editors include:

- storage locations to define raw and in-process locations where materials are consumed from or produced to
- products to define properties of discrete (unit-tracked) or process (lot-tracked) products produced from a route
- materials to define raw, intermediate or finished materials consumed or produced
- dispositions to control product routing, or hold scrap or release from hold reasons
- units of measure to define the measurement units for lots or units and associated conversions
- work schedules to define working and non-working periods for shifts and teams





working them

- data definitions to define data collected by an in-process lot or unit that can be used for quality analytics or other purposes
- reason code hierarchy to define loss and downtime reason codes for OEE availability tracking
- users and roles to define users and their groups used for local log in and team membership
- plugins to install, uninstall, enable and disable plugins that are considered to be outside of core MES functionality, but commonly needed in real world applications.
- settings to define property values that control how the clients behave

Work Cells, Equipment & Equipment Classes

The screenshot below shows the Functional Test Fixture equipment in the Functional Tester class from the SMT demo. It is using the Semi E10 state machine plugin.

The screenshot shows the MES AI Dashboard interface. The left sidebar contains a navigation menu with categories: PLANT MODEL (Sites, Equipment Classes, Storage Locations), PRODUCTS (Products, Routes, Dispositions, Materials), DEFINITIONS (Units of Measure, Work Schedules, Data Definitions, Reason Codes), ADMIN (Users, Roles, Plugins, Settings), and DEMOS (CPG Demo, Electronics Demo). The main content area is titled 'Equipment' and shows 'Equipment in work cell Functional Test Cell.' A search bar is present with the text 'Search by name or code...' and '1 item' next to it. Below the search bar is a table with the following data:

CODE	NAME	CLASS	STATE MODEL	ACTIVE	ACTIONS
FCT-200	Functional Test Fixture	Functional Tester (TESTER)	semi_e10	✓	   

Equipment can have capabilities that are used in the WIP dispatching algorithm:

[← Back to Equipment](#)

Equipment Capabilities

ISA-95 Part 2 formal capability declarations.

[+ Add Capability](#)

available Filler (FILLER) ✎ 🗑️

Reason: Nominal

Property	Value
max_fill_rate	300 bottle/min
min_fill_vol_ml	200 mL

It can also have material (WIP) production capabilities used in calculating the performance component of OEE:

[← Back to Equipment](#)

Material Capabilities

Define design speed, reject UoM, and target OEE for each material this equipment can produce.

[+ Add Material Capability](#)

Search by material name, code, or UoM... 1 capability

MATERIAL	CODE	DESIGN SPEED	SPEED UOM	REJECT UOM	TARGET OEE	ACTIONS
Premium Orange Juice 1 L	FG-OJ-1L	300	EA	EA	92%	✎ 🗑️

This is a screenshot of the Reflow Oven class editor with two class properties

MES AI
Dashboard

PLANT MODEL

- Sites
- Equipment Classes**
- Storage Locations

PRODUCTS

- Products
- Routes
- Dispositions
- Materials

DEFINITIONS

- Units of Measure
- Work Schedules
- Data Definitions
- Reason Codes

ADMIN

- Users
- Roles
- Plugins
- Settings

DEMOS

- CPG Demo
- Electronics Demo

← Back to Equipment Classes

Reflow Oven

Code: OVEN — Convection / IR reflow soldering oven (1 equipment assigned) [+ Add Property](#)

Assigned Equipment (1)

CODE	NAME	STATE MODEL	QUEUE DEPTH	ACTIVE
RO-500	5-Zone Reflow Oven	semi_e10	5	active

Class Properties

NAME	DATA TYPE	UOM	DEFAULT	DESCRIPTION	ACTIONS
max_temp_c	float	°C	260	Max zone temperature	✎ 🗑️
zone_count	int	—	5	Number of heating zones	✎ 🗑️

🔗 About MES AI
localhost:5175/equipment-classes

Sites, areas, production lines and work cells have some common properties like work schedule. Additionally a work cell can define the WIP dispatching strategy for all equipment that belongs to it. In this example dispatching is by capability match:

Edit Work Cell ✕

Code

Name

Description (optional)

Work Schedule (optional)

Manufacturing Company - four twelves
▾

Default Dispatch Strategy (optional)

Capability Match
▾

Used automatically when dispatch is evaluated without an explicit strategy.

Cancel

Update

Routes and Dispositions

Routes

The screenshot below shows the route editor for the SMT Assembly Line. This editor is used to create and manage manufacturing routes, steps, and material assignments.

Route Editor

Create and manage manufacturing routes, steps, and material assignments.

Routes

+ New

SMT Assembly Line

v1.0 ✎ 🗑

Juice Bottling Line

v1.0 ✎ 🗑

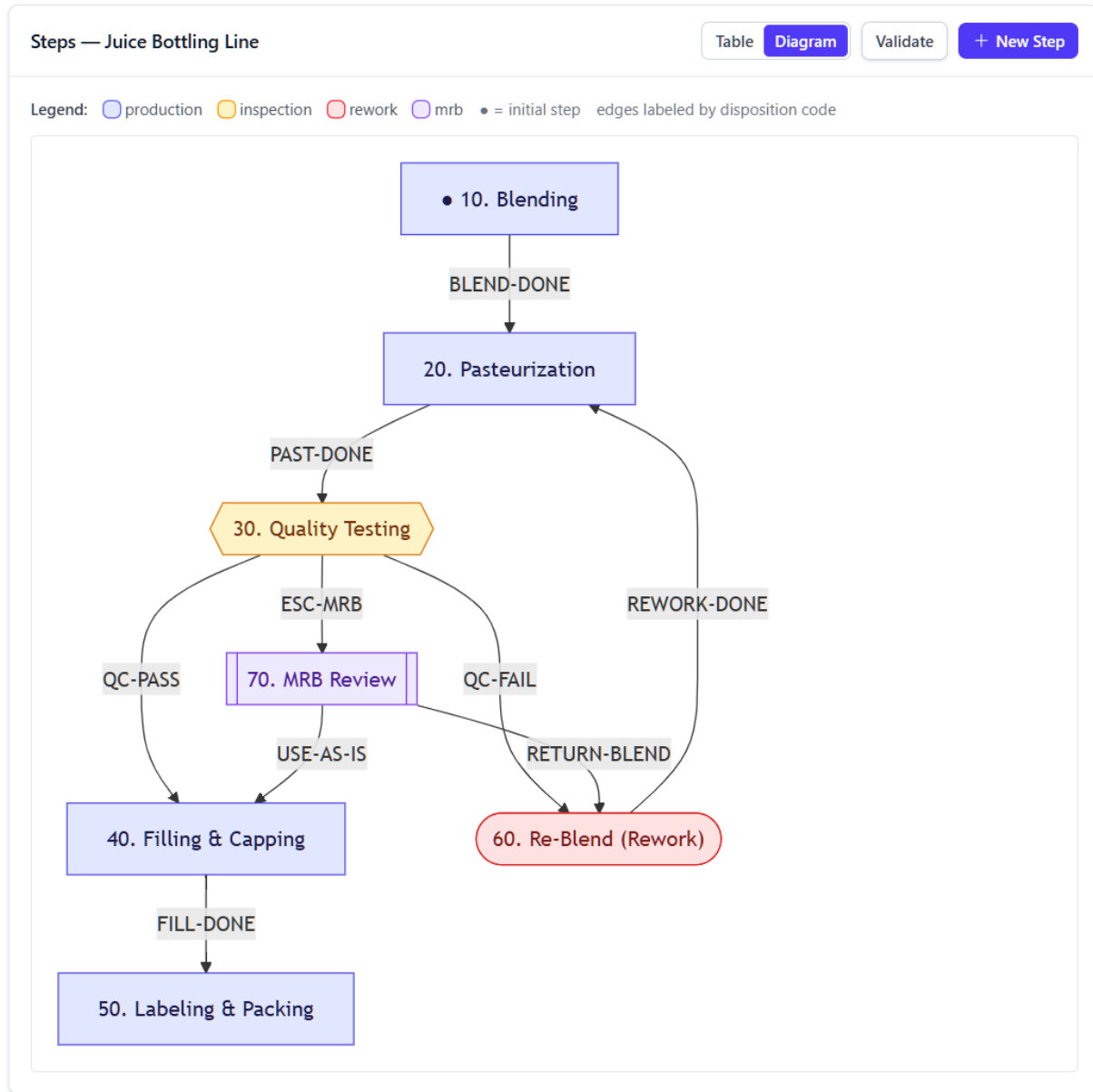
Steps — Juice Bottling Line

SEQ	NAME
10	Blending

Steps — Juice Bottling Line								
Table Diagram Validate + New Step								
SEQ	NAME	EQUIPMENT CLASS	HOSTED WORK CELLS	TYPE	INPUTS	OUTPUTS	CYCLE TIME	ACTIONS
10	Blending	MIXER	WC-BLEND	production INITIAL	—	BLEND-DONE	900s	
20	Pasteurization	PASTEURIZER +1	WC-PAST	production	BLEND-DONE REWORK-DONE	PAST-DONE	600s	
30	Quality Testing	ANALYZER +1	WC-QC	inspection	PAST-DONE	QC-PASS QC-FAIL ESC-MRB	300s	
40	Filling & Capping	FILLER +3	WC-FILL	production	QC-PASS USE-AS-IS	FILL-DONE	1200s	
50	Labeling & Packing	LABELER +1	WC-PACK	production	FILL-DONE	—	600s	
60	Re-Blend (Rework)	TANK +1	WC-REWORK	rework	QC-FAIL RETURN-BLEND	REWORK-DONE	600s	
70	MRB Review	ANALYZER +1	WC-QC	mrbs	ESC-MRB	RETURN-BLEND USE-AS-IS	600s	

Materials		+ Assign
<input checked="" type="radio"/> All <input type="radio"/> Products <input type="radio"/> Materials		
FG-OJ-1L	Premium Orange Juice 1 L process	
FG-OJ-1L	Premium Orange Juice 1 L finished	
PKG-CASE	Shipping Case (12-pack) packaging	
PKG-LABEL	Product Label packaging	
PKG-CAP	Bottle Cap packaging	
PKG-BOTTLE-1L	PET Bottle 1 L packaging	
SF-JUICE-BLEND	Blended Juice semi_finished	
RM-VITC	Vitamin C Powder raw	
RM-CITRIC	Citric Acid raw	
RM-SUGAR	Cane Sugar raw	
RM-WATER	Purified Water raw	
RM-OJ-CONC	Orange Juice Concentrate raw	

The steps can be viewed as a graph diagram:



When a step is created or selected in the table, the step editor is used to define step-related properties.

Edit Step



Sequence

10

Type

Production

Name

Blending

Equipment Requirements

Set the primary equipment class for the step and add any extra class or specific-equipment constraints. Dispatch ANDs them all.

Primary Equipment Class (ISA-95)

MIXER — Mixer

The step-level equipment class used by dispatch and shown in the route table.

No additional requirements. The primary equipment class above is the only constraint.

Equipment Class Specific Equipment

— Select class —

required


+ Add


Description (optional)

Material Requirements


Step-level BOM: materials consumed or produced at this segment.


RM-OJ-CONC — Orange Juice Conce... 200 kg consumed consumed 


RM-WATER — Purified Water 800 L consumed consumed 

RM-SUGAR — Cane Sugar 50 kg consumed consumed 

RM-CITRIC — Citric Acid 2 kg consumed consumed 

RM-VITC — Vitamin C Powder 0.5 kg consumed consumed 

SF-JUICE-BLEND — Blended Juice 1000 L produced produced 


— Select material — 


Quantity

UoM

Use

1

— UoM — 

consumed 

+ Add

Description (optional)

Step Parameters

Data-collection specs (measurements, set-points, attributes) captured at this step.

Blend Ratio target=4.0 [3.8, 4.2] numeric required  

Mix Temperature target=25 [20, 30] °C numeric required  

Mix Time target=15 [12, 20] min numeric required  

Add parameter

Name

Torque, Temperature, Serial #, ...

Data type

numeric

UoM

— none —

Target

setpoint

Lower limit

Upper limit

Required

+ Add

Data Definitions
 Catalog entries collected at this step (numeric readings, strings, booleans, enums — manual, equipment, or sensor). [Catalog](#)

No data definitions attached. Pick one from the catalog below or create a new one on the data-definition catalog page.

Attach existing catalog entry

— Select an unassigned data definition — + Attach

All catalog entries are already bound to a step. Create a new one on the [data-definition catalog](#).

Expected Cycle Time (sec) (optional)

900

ERP Operation # (optional)

0010

Initial step

Mark as the canonical entry point of the route. An initial step must have an empty input disposition list.

<p>Input Dispositions Dispositions that route WIP <i>into</i> this step. Only routing-category dispositions are available.</p> <ul style="list-style-type: none"> <input type="checkbox"/> E-PASTE-DONE — Paste Applied <input type="checkbox"/> FILL-DONE — Fill Complete <input type="checkbox"/> REWORK-DONE — Rework Complete <input type="checkbox"/> QC-FAIL — QC Fail <input type="checkbox"/> QC-PASS — QC Pass <input type="checkbox"/> PAST-DONE — Pasteurization Complete <input type="checkbox"/> BLEND-DONE — Blend Complete 	<p>Output Dispositions Dispositions that route WIP <i>out of</i> this step. Only routing-category dispositions are available.</p> <ul style="list-style-type: none"> <input type="checkbox"/> E-PASTE-DONE — Paste Applied <input type="checkbox"/> FILL-DONE — Fill Complete <input type="checkbox"/> REWORK-DONE — Rework Complete <input type="checkbox"/> QC-FAIL — QC Fail <input type="checkbox"/> QC-PASS — QC Pass <input type="checkbox"/> PAST-DONE — Pasteurization Complete <input checked="" type="checkbox"/> BLEND-DONE — Blend Complete
--	--

Cancel Update

In this example, step #10 (Blending) is the initial step on the juice bottling line and has only one output disposition "BLEND-DONE" which sends the lot to step #20 Pasteurization.

The equipment class panel states that equipment must be in the MIXER class to be

acceptable for dispatching a lot, and no specific equipment is required.

The material requirements panel defines five raw materials that are consumed and one intermediate material that is produced. The consumed materials are in the product Bill of Material (BOM). Click on the Products link in the sidebar, then on the name of the product to display the details editor:

← Products > FG-OJ-1L > BOMs

Bills of Material

Premium Orange Juice 1 L · FG-OJ-1L v1.0

BOMs + New BOM

Version 1.0
No effective date ✎ 🗑

Items for BOM v1.0					+ Add Item
POS	MATERIAL	QTY	UOM	CONSUMED AT STEP	
10	RM-OJ-CONC	200	kg	10. Blending	✎ 🗑
20	RM-WATER	800	L	10. Blending	✎ 🗑
30	RM-SUGAR	50	kg	10. Blending	✎ 🗑
40	RM-CITRIC	2	kg	10. Blending	✎ 🗑
50	RM-VITC	0.5	kg	10. Blending	✎ 🗑
60	PKG-BOTTLE-1L	1000	EA	40. Filling & Capping	✎ 🗑
70	PKG-CAP	1000	EA	40. Filling & Capping	✎ 🗑
80	PKG-LABEL	1000	EA	50. Labeling & Packing	✎ 🗑
90	PKG-CASE	84	EA	50. Labeling & Packing	✎ 🗑

The step parameters panel defines parameters at a route step than are typically setpoints or critical values. At runtime, the actual value can be collected.

The data definitions panel defines data that is not a step parameter to be collected at a route step. Here's an example from the SMT line at Functional Test:

Data Definitions
Catalog entries collected at this step (numeric readings, strings, booleans, enums — manual, equipment, or sensor). [Catalog](#)

ECB-FCT-FW Firmware Checksum OK	boolean	equipment	required	
ECB-FCT-IO I/O Channels OK	boolean	equipment	required	

Attach existing catalog entry

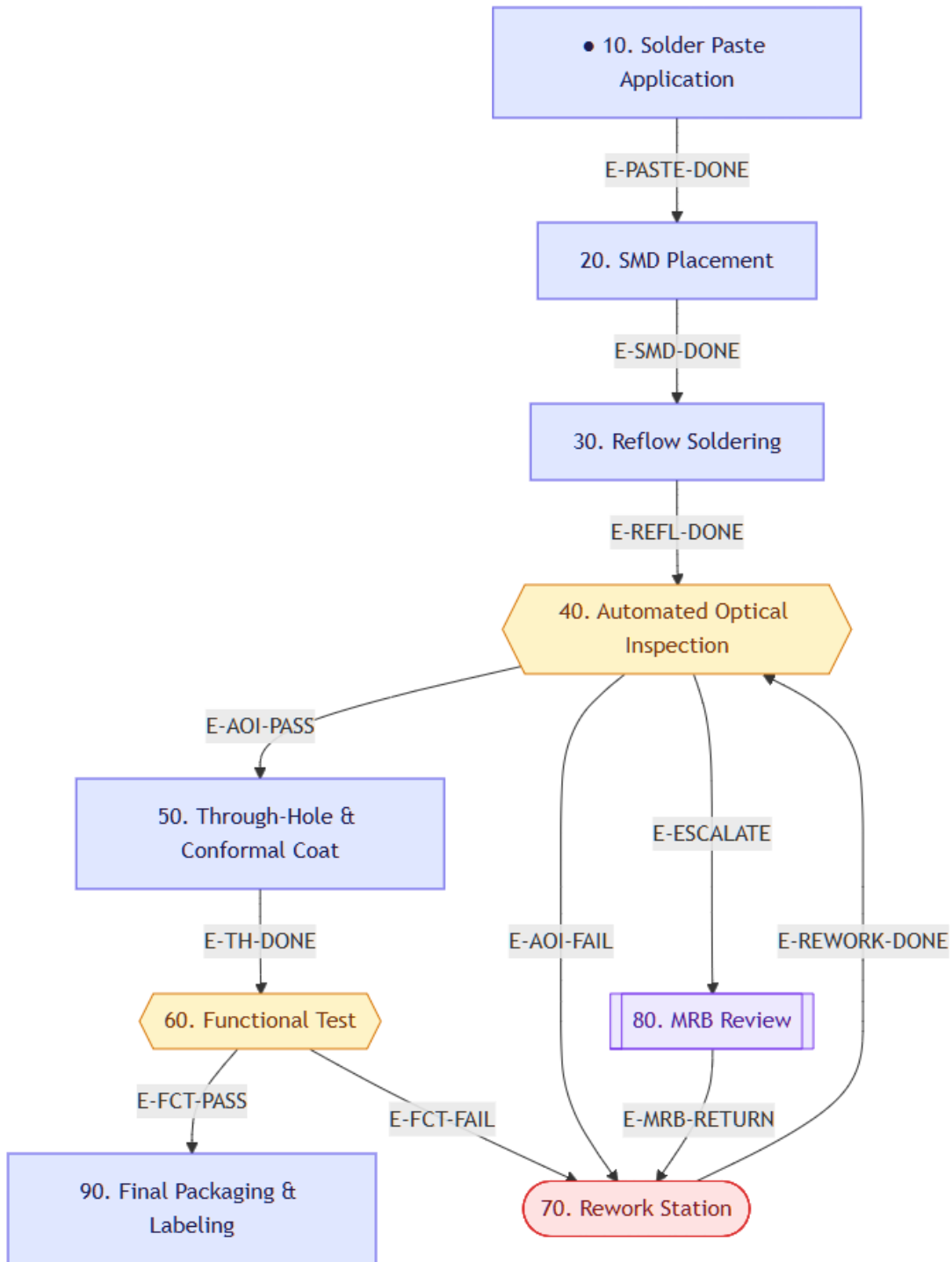
[+ Attach](#)

All catalog entries are already bound to a step. Create a new one on the [data-definition catalog](#).

and its route:

Steps — SMT Assembly Line

Legend: □ production ◡ inspection ◡ rework ▭ mrb ● = initial step edges labeled by disposition code



where the step #60 has one input disposition and two outputs:

marked as the canonical entry point of the route, its initial step must have an empty input disposition box.

Input Dispositions	Output Dispositions
Dispositions that route WIP <i>into</i> this step. Only routing-category dispositions are available.	Dispositions that route WIP <i>out of</i> this step. Only routing-category dispositions are available.
<input type="checkbox"/> E-REWORK-DONE — Rework Complete	<input type="checkbox"/> E-REWORK-DONE — Rework Complete
<input type="checkbox"/> E-FCT-FAIL — Functional Test Fail	<input checked="" type="checkbox"/> E-FCT-FAIL — Functional Test Fail
<input type="checkbox"/> E-FCT-PASS — Functional Test Pass	<input checked="" type="checkbox"/> E-FCT-PASS — Functional Test Pass
<input checked="" type="checkbox"/> E-TH-DONE — TH & Coat Done	<input type="checkbox"/> E-TH-DONE — TH & Coat Done
<input type="checkbox"/> E-AOI-FAIL — AOI Fail	<input type="checkbox"/> E-AOI-FAIL — AOI Fail
<input type="checkbox"/> E-AOI-PASS — AOI Pass	<input type="checkbox"/> E-AOI-PASS — AOI Pass
<input type="checkbox"/> E-REFL-DONE — Reflow Done	<input type="checkbox"/> E-REFL-DONE — Reflow Done

Dispositions

The disposition editor is used to define the dispositions that route WIP through processing steps or put it on hold, release or scrap it.

Dispositions

Disposition codes that route, hold, or scrap WIP at each process step.

Route

+ New Disposition

CODE	NAME	DESCRIPTION	CATEGORY	ACTIONS
BLEND-DONE	Blend Complete	Blending finished, advance to pasteurization	route	
E-AOI-FAIL	AOI Fail	AOI failed, send to rework	route	
E-AOI-PASS	AOI Pass	AOI passed, advance to TH & coat	route	
E-FCT-FAIL	Functional Test Fail	Functional test failed, send to rework	route	
E-FCT-PASS	Functional Test Pass	Functional test passed (terminal)	route	
E-PASTE-DONE	Paste Applied	Solder paste applied, advance to placement	route	
E-REFL-DONE	Reflow Done	Reflow complete, advance to AOI	route	
E-REWORK-DONE	Rework Complete	Rework station complete, return for re-inspection	route	
E-SMD-DONE	SMD Placed	SMD components placed, advance to reflow	route	
E-TH-DONE	TH & Coat Done	Through-hole and conformal coat done, advance to test	route	
FILL-DONE	Fill Complete	Filling/capping done, advance to packing	route	
PAST-DONE	Pasteurization Complete	Pasteurization finished, advance to QC	route	
QC-FAIL	QC Fail	Quality test failed — send to rework	route	
QC-PASS	QC Pass	Quality test passed	route	
REWORK-DONE	Rework Complete	Re-blend complete, return to pasteurization	route	

Products

Products are produced material. The products editor looks like this:

MES AI
Dashboard

PLANT MODEL

- Sites
- Equipment Classes
- Storage Locations

PRODUCTS

- Products**
- Routes

Products

Define product specifications — code, version, UoM, type.

Filter by type: All types 2 products

CODE	NAME	VERSION	TYPE	UOM	ACTIONS
FG-ECB-100	Electronic Controller Board v1	1.0	discrete	EA	
FG-OJ-1L	Premium Orange Juice 1 L	1.0	process	EA	

+ New Product

You can edit a product or delete it. Click on a product to view its route and step details as well as its Bill of Material:

Electronic Controller Board v1 Manage BOMs Active
FG-ECB-100 · v1.0 · discrete · EA

Process Routes + New Route

SMT Assembly Line v1.0 default > ✎ 🗑

Steps — SMT Assembly Line + New Step

SEQ	NAME	TYPE	CYCLE TIME	INPUTS	OUTPUTS	CONSUMABLES
10	Solder Paste Application +1	production	30s	—	E-PASTE-DONE	RM-PASTE RM-BLA
20	SMD Placement +3	production	45s	E-PASTE-DONE	E-SMD-DONE	RM- (1 EA)
30	Reflow Soldering +1	production	180s	E-SMD-DONE	E-REFL-DONE	—
40	Automated Optical Inspection	inspection	20s	E-REFL-DONE E-REWORK-DONE	E-AOI-PASS E-AOI-FAIL E-ESCALATE	—

Step Detail
10. Solder Paste Application initial

Input Dispositions (WIP routed in by)
Entry point — no input dispositions needed

Output Dispositions (WIP routed out by)
E-PASTE-DONE

1 output → **auto-routed**. No operator disposition pick needed.

✎ Edit Step

You can also edit its BOM, by clicking on the Manage BOM button:

Bills of Material

Electronic Controller Board v1 · FG-ECB-100 v1.0

BOMs

+ New BOM

Version 1.0

No effective date



Items for BOM v1.0

8 items

+ Add Item

POS	MATERIAL	QTY	UOM	CONSUMED AT STEP	
10	RM-PCB-BLANK	1	EA	10. Solder Paste Application	
20	RM-SMD-KIT	1	EA	20. SMD Placement	
30	RM-THRU-KIT	1	EA	50. Through-Hole & Conformal Coat	
40	RM-SOLDER-PST	5	g	10. Solder Paste Application	
50	RM-FLUX	2	mL	50. Through-Hole & Conformal Coat	
60	RM-CONFORMAL	3	mL	50. Through-Hole & Conformal Coat	
70	SF-POP-PCB	1	EA	—	
80	PKG-ESD-BAG	1	EA	90. Final Packaging & Labeling	

Materials

Materials are consumed or produced by equipment. The Materials editor allows a user to create, update and delete materials and search by type. For example:

MES AI
Dashboard

Materials
Define raw, intermediate, and finished material specifications.

+ New Material

PLANT MODEL

- Sites
- Equipment Classes
- Storage Locations

PRODUCTS

- Products
- Routes
- Dispositions
- Materials**

DEFINITIONS

- Units of Measure

Filter by type: All types 20 materials

CODE	NAME	TYPE	UOM	SHELF LIFE (DAYS)	ACTIONS
FG-ECB-100	Electronic Controller Board v1	finished	EA	-	
PKG-ESD-BAG	ESD Protective Bag	packaging	EA	-	
SF-POP-PCB	Populated PCB Assembly	semi_finished	EA	-	
RM-CONFORMAL	Conformal Coating	raw	mL	-	
RM-FLUX	Flux Solution	raw	mL	-	

Storage Locations

Storage locations are used for storing WIP. For example:

Storage Locations
Manage warehouse and plant storage locations for inventory management.

+ New Location

Search by name, code, or description... All Types 16 locations

CODE	NAME	TYPE	POSITION	SITE	CAPACITY	ACTIVE	ACTIONS
LOC-01-01	A location	Storage	1 / 1 / 2	Apex Electronics Factory	70	✓	
EB-SHIP-01	Finished Goods Shipping	Shipping	—	Apex Electronics Factory	—	✓	
EB-RIP-SMT	RIP — SMT Line	Raw-in-Process	—	Apex Electronics Factory	—	✓	
EB-STG-01	Line-Side Staging	Staging	—	Apex Electronics Factory	—	✓	
EB-WH-PKG	Packaging Warehouse	Storage	B / 01 / 01	Apex Electronics Factory	—	✓	
EB-WH-SMT	SMT Component Warehouse	Storage	A / 01 / 01	Apex Electronics Factory	—	✓	
EB-RECV-01	Electronics Receiving Dock	Receiving	—	Apex Electronics Factory	—	✓	
SB-SHIP-01	Shipping Dock 1	Shipping	—	Sunrise Beverages	—	✓	
SB-RIP-PACK	RIP — Packing	Raw-in-Process	—	Sunrise Beverages	—	✓	
SB-RIP-FILL	RIP — Filling	Raw-in-Process	—	Sunrise Beverages	—	✓	
SB-RIP-BLEND	RIP — Blending	Raw-in-Process	—	Sunrise Beverages	—	✓	
SB-STG-01	Staging Area 1	Staging	—	Sunrise Beverages	—	✓	
SB-WH-B01	Packaging Stock B-01-01	Storage	B / 01 / 01	Sunrise Beverages	—	✓	
SB-WH-A02	Raw Dry Goods A-01-02	Storage	A / 01 / 02	Sunrise Beverages	—	✓	
SB-WH-A01	Raw Liquids A-01-01	Storage	A / 01 / 01	Sunrise Beverages	—	✓	
SB-RECV-01	Receiving Dock 1	Receiving	—	Sunrise Beverages	—	✓	

Units of Measure

Units of measure are paired with a numerical quantity of a substance. They have a type (time, length, mass, temperature, etc.) and a class - scalar (e.g. kilogram), quotient (e.g. meters/sec), power (square meters) or product (kg * m). Complex types like a Newton can be created from intermediate UOMs.

Units of Measure

Define units, conversion factors, and composite units.

[+ New Unit](#)

Filter by type: 76 units

SYMBOL	NAME	TYPE	CLASS	FORMULA	MULTIPLIER	OFFSET	BUILT-IN	ACTIONS
kgm	kg*m	Custom	product	kg x m	-	-	-	
in ³	cubic inch	Length	power	in ³	-	-	✓	
in ²	square inch	Length	power	in ²	-	-	✓	
ft ³	cubic foot	Length	power	ft ³	-	-	✓	
ft ²	square foot	Length	power	ft ²	-	-	✓	
mm ³	cubic millimeter	Length	power	mm ³	-	-	✓	
mm ²	square millimeter	Length	power	mm ²	-	-	✓	
cm ³	cubic centimeter	Length	power	cm ³	-	-	✓	
cm ²	square centimeter	Length	power	cm ²	-	-	✓	
m ³	cubic meter	Length	power	m ³	-	-	✓	
m ²	square meter	Length	power	m ²	-	-	✓	
PC/h	pieces per hour	Count	quotient	PC ÷ h	-	-	✓	
EA/h	each per hour	Count	quotient	EA ÷ h	-	-	✓	
EA/min	each per minute	Count	quotient	EA ÷ min	-	-	✓	

Compatible types can be converted from one to another. For example:

Conversion Test

Type: Value: From: To: [Convert](#)

1 km = 3280.8398950131 ft (kilometer → foot)

Data Definitions

A data definition is a specification of a data value to be collected during WIP processing. For example:

Data Definitions

Define data collection points — what to measure, limits, and validation rules.

[+ New Definition](#)

Type: Source: 3 definitions

CODE	NAME	DATA TYPE	SOURCE	UOM	LIMITS	REQ	ACTIONS
ECB-FCT-FW	Firmware Checksum OK	boolean	equipment	—	—	✓	✎ 🗑
ECB-FCT-IO	I/O Channels OK	boolean	equipment	—	—	✓	✎ 🗑
ECB-AOI-DEF	Defect Count	numeric	equipment	count	0 - 3	✓	✎ 🗑

Reason Codes

Reason codes are used to provide the OEE availability bucket when an equipment event occurs. For example:

Reason Codes

Hierarchical loss and downtime reason codes for OEE availability tracking.

[+ New Reason](#)

CODE	NAME	DESCRIPTION	OEE BUCKET	ACTIONS
0000	Running	Running normally	Value-Add	+ ✎ 🗑
1000	Mechanical	Mechanical reasons	Excluded	+ ✎ 🗑
1010	EStop	Emergency stop	Unplanned DT	+ ✎ 🗑
1020	Motor Overload	Motor above temp	Unplanned DT	+ ✎ 🗑
2000	Electrical	Electrical reasons	Excluded	+ ✎ 🗑
2010	GFI	GFI trip	Unplanned DT	+ ✎ 🗑
3000	Maintenance	Maintenance reasons	Excluded	+ ✎ 🗑
3010	PM	Preventive maintenance	Planned DT	+ ✎ 🗑
3020	Meeting	Team meeting	Non-Value	+ ✎ 🗑

Users and Roles

These two editors are used to create, edit and delete users and roles when managing authentication in the local database. For example:

MES AI
Dashboard

PLANT MODEL

- Sites
- Equipment Classes
- Storage Locations

PRODUCTS

- Products
- Routes
- Dispositions
- Materials

DEFINITIONS

- Units of Measure
- Work Schedules
- Data Definitions
- Reason Codes

ADMIN

- Users**

Users

Manage MES user accounts and role assignments. + New User

USERNAME	FULL NAME	EMAIL	ROLES	LAST LOGIN	
admin	MES Administrator	admin@mes.local	admin	—	
cpg_engineer	Emma García	cpg.engineer@mes.local	engineer	—	
cpg_operator1	Tom Williams	cpg.operator1@mes.local	operator	—	
cpg_operator2	Sara Kim	cpg.operator2@mes.local	operator	—	
plant_manager	Alex Johnson	plant.manager@mes.local	viewer	—	
smt_engineer	David Chen	smt.engineer@mes.local	engineer	—	
smt_operator1	Maria Rossi	smt.operator1@mes.local	operator	—	
smt_operator2	James Park	smt.operator2@mes.local	operator	—	
sqa-team-user-1d48815f	SQA Team User	—	none	—	

Roles

Manage roles and their permission sets.

+ New Role

NAME	DESCRIPTION	PERMISSIONS	TYPE	
admin	System administrator with full access	1 permission	System	
engineer	Process/manufacturing engineer	9 permissions	System	
operator	Shop floor operator	11 permissions	System	
viewer	Read-only access for management and auditors	1 permission	System	
super user	A user with all privileges	0 permissions	Custom	

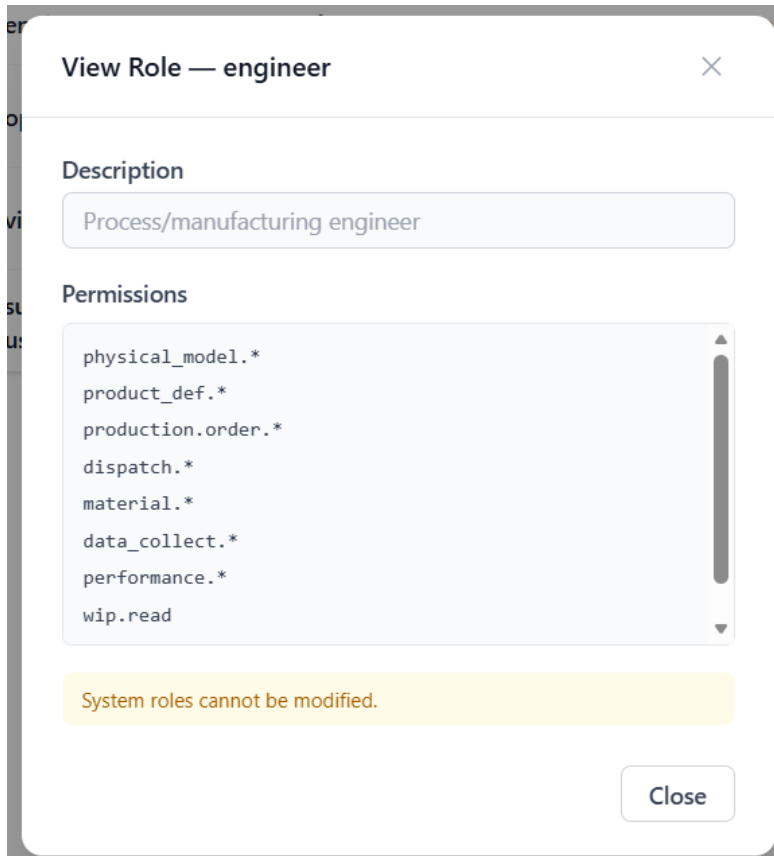
Permissions in the roles editor are dot-separated strings (e.g. material.read) that are

stored on a Role and checked server-side via the `require_permission()` FastAPI dependency on every protected route. When a user makes an API call, their aggregated permissions are matched against the required string — including wildcard support (*, `wip.*`, `*.read`).

The permission namespace is divided by domain and action:

Domain	Actions observed
auth.user	read, create, update, delete
auth.role	read, create, update, delete
physical_model	read, create, update, delete
product_def	read, create, update, delete
material	read, create, update, delete
wip	read, create + sub-actions like <code>wip.unit.move</code>
production	read, create, update
performance	read, create
inventory	read, create, update
data_collection	read, create
dispatch	read, execute
genealogy	read

For example:




Work Schedule

This editor defines the shifts worked by teams as well as non-working time (i.e. holidays). Work schedules are assigned to levels in the S95 hierarchy, for example by site. A lower=level schedule overrides a level higher u.

This schedule is four alternating 12 hours day/night shifts with 7 days on followed by 7 days off. There are two day A/C teams and two night B/D teams.

← Manufacturing Company - four twelves


Four 12 hour alternating day/night shifts

 Shift Instances

Shifts Rotations Teams Non-Working Periods

+ New Shift

▼ **Day** Day shift



07:00 12h 0m 0 breaks  

BREAKS

+ Add Break

No breaks.

> **Night** Night shift

19:00 12h 0m 0 breaks  

← Manufacturing Company - four twelves



Four 12 hour alternating day/night shifts

 Shift Instances

Shifts Rotations Teams Non-Working Periods

+ New Rotation

▼ **Day** Day



14 days  

SEGMENTS

+ Add Segment

#1 **Day** 7 on / 7 off 

> **Night** Night

14 days  



← Manufacturing Company - four twelves

Four 12 hour alternating day/night shifts


📄 Shift Instances



Shifts Rotations Teams Non-Working Periods



+ New Team



▼ **A** A day shift starts 2014-01-02 1 member  

MEMBERS + Add Member

Tom Williams `cpg_operator1` 

> **B** B night shift starts 2014-01-02 0 members  

> **C** C day shift starts 2014-01-09 0 members  

> **D** D night shift starts 2014-01-09 0 members  

Once a work schedule is defined, the shift instances for a period of time can be calculated:

Shift Instances — Manufacturing Company - four twelves



From

10/01/2026 07:00 AM



To

10/29/2026 07:00 AM



Show Shifts

Working time: 672 h 0 min

Non-working time: 0 h 0 min

Shift	Start	End	Team
Day	10/01/2026, 07:00 AM	10/01/2026, 07:00 PM	C
Night	10/01/2026, 07:00 PM	10/02/2026, 07:00 AM	D
Day	10/02/2026, 07:00 AM	10/02/2026, 07:00 PM	C
Night	10/02/2026, 07:00 PM	10/03/2026, 07:00 AM	D
Day	10/03/2026, 07:00 AM	10/03/2026, 07:00 PM	C
Night	10/03/2026, 07:00 PM	10/04/2026, 07:00 AM	D
Day	10/04/2026, 07:00 AM	10/04/2026, 07:00 PM	C
Night	10/04/2026, 07:00 PM	10/05/2026, 07:00 AM	D
Day	10/05/2026, 07:00 AM	10/05/2026, 07:00 PM	C
Night	10/05/2026, 07:00 PM	10/06/2026, 07:00 AM	D
Day	10/06/2026, 07:00 AM	10/06/2026, 07:00 PM	C
Night	10/06/2026, 07:00 PM	10/07/2026, 07:00 AM	D
Day	10/07/2026, 07:00 AM	10/07/2026, 07:00 PM	C
Night	10/07/2026, 07:00 PM	10/08/2026, 07:00 AM	D
Day	10/08/2026, 07:00 AM	10/08/2026, 07:00 PM	A
Night	10/08/2026, 07:00 PM	10/09/2026, 07:00 AM	B
Day	10/09/2026, 07:00 AM	10/09/2026, 07:00 PM	A

Close

Plugins

Plugins are used to extend functionality beyond the core MES WIP, inventory and equipment tracking. A plugin is implemented with an adapter which interfaces to the REST API, and typically a simulator client for developing and testing purposes. The adapter typically has a configuration editor.

A plugin is first installed from the available list, then enabled. The plugin editor has a tab for installed ones and available ones. For example:

Plugins













Browse available plugins, install, and manage their lifecycle.

Installed (4)

Available (16)

Search by name, ID, or category...

4 plugins

NAME	VERSION	ORIGIN	CATEGORY	STATUS	ACTIONS
Oracle Cloud ERP Simulator oracle-erp-simulator	1.0.0	system	erp	running	  
PackML Equipment State Model packml-availability	1.0.0	system	performance	running	  
PackML OPC-UA Production Counters packml-opcua-counters	1.0.0	system	performance	running	  
SEMI E10 Equipment State Model semi-e10-availability	1.0.0	system	performance	running	  

Plugins






Browse available plugins, install, and manage their lifecycle.

Installed (4)

Available (16)

Search by name, ID, or category...

16 plugins

NAME	VERSION	ORIGIN	CATEGORY	STATUS	ACTIONS
AVEVA Historian Equipment Adapter aveva-historian	1.0.0	system	equipment	available	 
Equipment Simulator equipment-simulator	1.0.0	system	performance	available	 
Example Dispatch Plugin example-dispatch-plugin	0.1.0	system	dispatch	available	 
File-Drop Test Results Collector file-drop-test-results	1.0.0	system	data-collection	available	 
Mock Equipment Adapter mock-equipment	1.0.0	system	equipment	available	 
Mock ERP Adapter mock-erp	1.0.0	system	erp	available	 
Mock Test Equipment Adapter mock-test-equipment	1.0.0	system	test-equipment	available	 
Modbus TCP/RTU Equipment Adapter modbus-equipment	1.0.0	system	equipment	available	 
Modbus Equipment Simulator modbus-equipment-simulator	1.0.0	system	equipment	available	 
MQTT Production Counter Collector mqtt-counters	1.0.0	system	performance	available	 
MQTT Equipment Adapter mqtt-equipment	1.0.0	system	equipment	available	 
OPC-UA Equipment Adapter opcua-equipment	1.0.0	system	equipment	available	 

Enterprise Plugins

The MES framework implements two ERP plugins: SAP and Oracle. For example,

here is the Oracle adapter configuration:

← Oracle Cloud ERP Adapter

oracle-cloud-erp v1.0.0 · MES AI Contributors · system

running Disable Uninstall

DESCRIPTION Oracle Cloud ERP (Fusion) integration via REST APIs. Pulls work orders, items, products, BOMs, routings, and work centers. Pushes completions, material transactions, scrap, labor, and quality results.	COMMENT Requires Oracle Cloud ERP with REST APIs enabled. Configure via MES_ERP_* and MES_ORACLE_* environment variables. Install with: pip install mes-ai[oracle]
CATEGORY erp	MIN MES VERSION 0.1.0
EXTENSION POINTS erp_inbound, erp_outbound	EVENT SUBSCRIPTIONS —
DEPENDENCIES —	REQUIRED PERMISSIONS —

Configuration

Base URL * <small>Oracle Cloud ERP base URL</small>	<input type="text" value="http://delphi/v1"/>
Authorization Type <small>Authentication type: oauth2, basic, or api_key</small>	<input type="text" value="api_key"/>
API Key Name <small>API key identifier (often unused; leave blank if not required)</small>	<input type="text" value="my_key"/>
API Key <small>API key value sent in the auth header</small>	<input type="password" value="....."/>
Organization Code * <small>Oracle manufacturing organization code</small>	<input type="text" value="M1"/>

[Save Configuration](#)

and the corresponding simulator configuration:

← Oracle Cloud ERP Simulator

oracle-erp-simulator v1.0.0 · MES AI Contributors · system

running

Disable

Uninstall

DESCRIPTION

Realistic Oracle Cloud ERP (Fusion) simulator for development and integration testing. Generates Oracle-authentic REST API messages using real Oracle field names (WorkOrderNumber, ItemNumber, OrganizationCode, etc.) and transforms them through the OracleTransformLayer. Inbound: inventory items, item structures (BOMs), work orders, routings, work centers. Outbound: accepts completion transactions, material issue transactions, and returns Oracle-style transaction numbers.

COMMENT

Use this plugin when developing or testing the MES against Oracle Cloud ERP without a live Oracle Fusion instance. Unlike generic mock plugins, this simulator produces data in native Oracle REST format and exercises the full OracleTransformLayer pipeline.

CATEGORY

erp

MIN MES VERSION

0.1.0

EXTENSION POINTS

erp_inbound, erp_outbound

EVENT SUBSCRIPTIONS

—

DEPENDENCIES

—

REQUIRED PERMISSIONS

—

Configuration

Organization Code

Oracle Inventory Organization Code

Business Unit

Oracle Business Unit name

Latency Ms

Simulated REST API latency in milliseconds per call

Failure Rate

Probability of simulated Oracle REST API errors (0.0 to 1.0)

Save Configuration

Only one ERP plugin should be enabled, and either the adapter or the simulator.

Factory Floor Plugins

This MES framework provides 12 factory floor plugins. The equipment related ones for performance, availability and quality are:

- PackML Equipment State Model
- PackML OPC-UA Production Counters
- SEMI E10 Equipment State Model
- Equipment Simulator
- AVEVA Historian Equipment Adapter
- Modbus TCP/RTU Equipment Adapter and Modbus Equipment Simulator
- MQTT Equipment Adapter and MQTT Production Counter Collector
- OPC-UA Equipment Adapter

- STOMP Equipment Adapter for JMS messages
- STOMP Equipment Adapter
- Example Dispatch Plugin for a custom dispatch algorithm template

and for parsing test result data files:

- File-Drop Test Results Collector

Mock Plugins

The mock plugins provide a template for further development of a plugin of the associated type.

Settings

The settings editor is used to change default MES properties in server, event bus and logging categories.

Runtime Client (rt-client)

rt-client is the runtime application for creating and processing WIP, dispatching work, recording inventory activity, and handling shop-floor execution. It has 6 tabs:

Dashboard, WIP, Orders, Inventory, Equipment and Live Events.

Dashboard

The screenshot shows the 'MES Runtime — Shop Floor' dashboard. It features a navigation bar with tabs for Dashboard, WIP, Orders, Inventory, Equipment, and Live Events. The main content area is titled 'Shop Floor Dashboard' and includes a 'Refresh' button. Below this, there are two main sections: 'Active Order Progress' and 'Recent Events'.

Active Order Progress

Order #	Status	Ordered	Completed	Scrapped	WIP	Throughput
SQA-ECB-20260519193830	In progress	1	0	0	2 queued 1 active	0/1

Recent Events

5:47:53 PM wip.unit.started ("unit_id": "f814b4e9-2c90-4d9b-b5ce-b1650aad90c9", "step_id": "08933918-6ce4-4dd5-912a-79f65773276e", "equipment_id": "e8b02b2f-5ab1-49f5-8b1...

The Dashboard shows the progress of released orders and recent WIP events, e.g. start and complete transactions.

Dispatch

Dispatch answers one question: given a unit or lot queued at a route step, which piece of equipment should process it?

Trigger

Dispatch can be invoked three ways:

- Manual — a call to POST /dispatch/evaluate, reviews ranked options, then calls POST /dispatch/execute
- Auto — POST /dispatch/auto evaluates with the specified strategy and immediately executes the top recommendation
- Event-driven — a WIP completion event (OPC-UA, MQTT, etc.) triggers auto_dispatch via the event handler

Evaluation Pipeline

1. Resolve WIP — load the unit (by serial number) or lot (by lot number); read its current_step_id
2. Find candidate equipment at the current step using ISA-95 priority:
 - SegmentEquipmentRequirement rows → specific named equipment for this step
 - equipment_class_id on the step → all equipment belonging to that class
 - Neither → no options, dispatch is not applicable
3. Resolve strategy — use the strategy passed in the request, or fall back to the work cell's default_dispatch_strategy, or default to first_available
4. Filter candidates — three hard gates, in order:

Gate	Condition	Result if failed
Availability	dispatch_category == "available" (derived from PackML/SEMI E10 state)	excluded
Capability	EquipmentMaterial row exists for the WIP's material	excluded
Capacity	queue_depth < max_queue_depth (if limit set)	excluded

5. Rank eligible options by strategy (see below)
6. Return ranked options + recommended (top result) + excluded_options with reasons. If no eligible equipment: blocked=True

Strategies

Strategy	Behavior
first_available	Return equipment in natural (database) order; first one wins
shortest_queue	Sort by ascending queue_depth; routes to least-loaded equipment
round_robin	Also sorts by queue_depth as a least-recently-used proxy
capability_match	Currently treats all eligible equipment as capable (placeholder for future capability property matching)
manual	No ranking; all options returned with score 0 for operator selection
custom	Natural-language prompt stored on the work cell (WorkCell.custom_strategy_prompt); currently keyword-matched (e.g. "shortest" → shortest_queue, "balance" → round_robin); designed to be replaced with an LLM call

For capability match, a production system would need to complete the algorithm.

- Load the EquipmentCapability rows for each candidate equipment where capability_type == "available" (and time bounds cover now)
- Compare the capability's property values against the step's requirements (e.g. SegmentEquipmentRequirement or a future SegmentCapabilityRequirement table) — e.g. "this step needs max_fill_rate ≥ 400; FL-400A declares 300 → exclude"
- Score/rank survivors by how well their declared property values satisfy the step requirements

Execution

execute() re-validates availability, capacity, and material setup, checks that the target step still matches the WIP's current_step_id (prevents stale dispatches from moving WIP), then sets unit.current_equipment_id / lot.current_equipment_id. Step advancement is not touched.

Status Signals

- Blocked — no equipment passed all three gates; dispatch_blocked event emitted
- Starved — equipment is available but queue_depth == 0; equipment_starved event emitted
- At capacity — queue_depth >= max_queue_depth

WIP

The WIP tab has two sub-tabs: Scan WIP and Active WIP. On the scan tab, a unit's serial number or lot's lot number can be entered/scanned to display information about this WIP. In the screen shot below, the UI shows an SMT unit that is started at AOI.

The screenshot displays the 'Scan WIP' interface. At the top, there are tabs for 'Scan WIP' and 'Active WIP'. Below the tabs, there is a search bar with a 'Serial Number' field containing 'SQA-ECB-SN-20260519193830' and a 'Search' button. A 'New Scan' link is visible in the top right corner. The main content area shows the unit ID 'SQA-ECB-SN-20260519193830' with a status indicator 'in process'. Below this is a 'Step History' table with columns for Seq, Step, Started, Completed, and Result. The final row shows a step 'Automated Optical Inspection' completed on 5/20/2026. At the bottom, there is an 'Equipment at Step' table showing 'AOI-300 (assigned)' is 'available' with a queue of 1/1 and capacity 'Yes'.

Scan WIP Active WIP

Scan WIP [New Scan](#)

Type Unit (Serial #) Lot (Lot #) Serial Number SQA-ECB-SN-20260519193830 Search

UNIT SQA-ECB-SN-20260519193830 in process

Step History

Seq	Step	Started	Completed	Result
10	Solder Paste Application	5/19/2026, 12:38:35 PM	5/19/2026, 12:38:37 PM	pass
20	SMD Placement	5/19/2026, 12:38:39 PM	5/19/2026, 12:38:41 PM	pass
30	Reflow Soldering	5/19/2026, 12:38:42 PM	5/19/2026, 12:38:44 PM	pass
40	Automated Optical Inspection	5/19/2026, 12:38:46 PM	5/19/2026, 12:38:48 PM	pass
50	Through-Hole & Conformal Coat	5/19/2026, 12:38:50 PM	5/19/2026, 12:38:53 PM	pass
60	Functional Test	5/19/2026, 12:38:55 PM	5/19/2026, 12:38:56 PM	pass
70	Rework Station	5/19/2026, 12:38:58 PM	5/19/2026, 12:38:59 PM	pass
40	Automated Optical Inspection	5/20/2026, 5:47:53 PM	—	—

Equipment at Step

Equipment	Availability	State	Queue	Capacity	Material
AOI-300 (assigned)	available	— (packml)	1 / 1	Yes	✓

Data Collection
Save

Defect Count * (count)

Spec: 0-3

Step Parameters

Parameter	Target	Lower	Upper	UoM	Actual
Resolution *	15	10	20	µm	<input style="width: 50px;" type="text" value="16"/>
Inspection Time *	8	5	15	s	<input style="width: 50px;" type="text" value="8"/>
Defect Threshold *	0	0	3	count	<input style="width: 50px;" type="text" value="1"/>

Complete Step

Disposition

AOI Pass → 50: Through-Hole & Conformal Coat — AOI passed, advance to TH & coat

Transition State

Place on Hold

Scrap

At this step, there is defect count to be collected and three step parameters to record. Clicking the Complete button will complete the unit at AOI to the step determined by the selected disposition.. If the equipment is being monitored in a state model (PackML or Semi-E10), checking the "Transition State" checkbox will first transition the equipment to the proper state before doing a start or complete transaction in order to avoid a runtime error.

The unit can also be placed on hold or scrapped here.

The Active WIP tab provides status filtering to search for units or lots. For example, for CPG lots:

Active WIP

[Refresh](#)

View
 Units Lots All

Lot #	Qty	Status	Current Step	Order	Created	
SQA-MRB-OJ-20260519193729	100 EA	scrapped	Quality Testing	SQA-MRB-20260519193729	5/19/2026, 12:37:32 PM	Open Report
SQA-OJ-20260519193700	100 EA	completed	—	SQA-WIP-20260519193700	5/19/2026, 12:37:04 PM	Open Report
SQA-MRB-OJ-20260516220906	100 EA	scrapped	Quality Testing	SQA-MRB-20260516220906	5/16/2026, 3:09:09 PM	Open Report
SQA-OJ-20260516220838	100 EA	completed	—	SQA-WIP-20260516220838	5/16/2026, 3:08:42 PM	Open Report
SQA-MRB-OJ-20260516192744	100 EA	scrapped	Quality Testing	SQA-MRB-20260516192744	5/16/2026, 12:27:47 PM	Open Report
SQA-OJ-20260516192713	100 EA	completed	—	SQA-WIP-20260516192713	5/16/2026, 12:27:18 PM	Open Report
SQA-MRB-OJ-20260515231524	100 EA	scrapped	Quality Testing	SQA-MRB-20260515231524	5/15/2026, 4:15:28 PM	Open Report
SQA-OJ-20260515231450	100 EA	completed	—	SQA-WIP-20260515231450	5/15/2026, 4:14:53 PM	Open Report
SQA-MRB-OJ-20260515230902	100 EA	scrapped	Quality Testing	SQA-MRB-20260515230902	5/15/2026, 4:09:07 PM	Open Report
SQA-OJ-20260515230827	100 EA	completed	—	SQA-WIP-20260515230827	5/15/2026, 4:08:31 PM	Open Report
SQA-MRB-OJ-20260515223939	100 EA	scrapped	Quality Testing	SQA-MRB-20260515223939	5/15/2026, 3:39:43 PM	Open Report
SQA-OJ-20260515223908	100 EA	completed	—	SQA-WIP-20260515223908	5/15/2026, 3:39:12 PM	Open Report
SQA-MRB-OJ-20260515210025	100 EA	scrapped	Quality Testing	SQA-MRB-20260515210025	5/15/2026, 2:00:29 PM	Open Report
SQA-OJ-20260515205956	100 EA	completed	—	SQA-WIP-20260515205956	5/15/2026, 2:00:00 PM	Open Report
SQA-MRB-OJ-20260515205900	100 EA	scrapped	Quality Testing	SQA-MRB-20260515205900	5/15/2026, 1:59:05 PM	Open Report
SQA-OJ-20260515205827	100 EA	completed	—	SQA-WIP-20260515205827	5/15/2026, 1:58:31 PM	Open Report

Clicking the Open link shows the lot step history:

Scan WIP **Active WIP**

[Back to list](#)

LOT

SQA-OJ-20260519193700 completed

Quantity: 100 EA

Step History

Seq	Step	Started	Completed	Result
10	Blending	5/19/2026, 12:37:06 PM	5/19/2026, 12:37:11 PM	pass
20	Pasteurization	5/19/2026, 12:37:12 PM	5/19/2026, 12:37:14 PM	pass
30	Quality Testing	5/19/2026, 12:37:16 PM	5/19/2026, 12:37:18 PM	pass
40	Filling & Capping	5/19/2026, 12:37:20 PM	5/19/2026, 12:37:22 PM	pass
50	Labeling & Packing	5/19/2026, 12:37:24 PM	5/19/2026, 12:37:27 PM	pass

All steps completed

Clicking the Report link produces a WIP History Report as a pdf that can be printed or saved. For example:

WIP History Report

Generated: 5/21/2026, 10:50:44 AM

WIP:	Lot — #SQA-OJ-20260519193700	Status:	completed
Order:	SQA-WIP-20260519193700	Product:	Premium Orange Juice 1 L

Processing History (5 steps)

Step	Equipment	Started	Completed	Duration	Result
10 — Blending	Tank Mixer TM-100	5/19/2026, 12:37:06 PM	5/19/2026, 12:37:11 PM	5s	Pass
Mix Time			15		
Mix Temperature			25		
Blend Ratio			4		
20 — Pasteurization	HTST Pasteurizer PS-200	5/19/2026, 12:37:12 PM	5/19/2026, 12:37:14 PM	2s	Pass
HTST Temperature			72		
Hold Time			15		
Exit Temperature			4		
30 — Quality Testing	Lab Analyzer QC-300	5/19/2026, 12:37:16 PM	5/19/2026, 12:37:18 PM	2s	Pass
Brix			11.5		
pH			3.8		
Microbial Count			0		
Color Index			35		
Taste Approved			true		
40 — Filling & Capping	Filler FL-400B	5/19/2026, 12:37:20 PM	5/19/2026, 12:37:22 PM	2s	Pass
Fill Volume			1000		
Cap Torque			1.2		
Headspace			15		
50 — Labeling & Packing	Labeler/Packer LP-500	5/19/2026, 12:37:24 PM	5/19/2026, 12:37:27 PM	3s	Pass
Label Aligned			true		
Date Code Legible			true		
Case Count			12		

Materials Consumed (9)

Code	Name	Lot #	Qty Consumed	Consumed At
RM-OJ-CONC	Orange Juice Concentrate	OJC-2026-001	20	5/19/2026, 12:37:08 PM
RM-WATER	Purified Water	H2O-2026-001	80	5/19/2026, 12:37:09 PM
RM-SUGAR	Cane Sugar	SUG-2026-001	5	5/19/2026, 12:37:09 PM
RM-CITRIC	Citric Acid	CIT-2026-001	0.2	5/19/2026, 12:37:10 PM
RM-VITC	Vitamin C Powder	VTC-2026-001	0.05	5/19/2026, 12:37:10 PM
PKG-BOTTLE-1L	PET Bottle 1 L	BTL-2026-001	100	5/19/2026, 12:37:22 PM

For queued WIP, the dispatch algorithm can be evaluated. First find the WIP of interest then click the Dispatch button, e.g.:

Scan WIP **Active WIP** Refresh

Active WIP

View Status
 Units Lots Queued

Serial #	Status	Current Step	Order	Created	
SN-SQA-ECB-20260519193830-00003	queued	—	SQA-ECB-20260519193830	5/20/2026, 5:31:46 PM	Open Dispatch Report
SN-SQA-ECB-20260519193830-00002	queued	—	SQA-ECB-20260519193830	5/20/2026, 5:31:45 PM	Open Dispatch Report
SQA-ECB-SN-20260519193830	queued	Rework Station	SQA-ECB-20260519193830	5/19/2026, 12:38:33 PM	Open Dispatch Report

A dialog will be displayed listing the available strategies. The Evaluate button executes the logic and displays the result. For example:

Dispatch Evaluation ✕

Lot: LOT-FG-OJ-1L-MPG9VTWD-001-0001

Strategy

Shortest Queue ▶ Evaluate

EQUIPMENT	WORK CELL	STATE	MATERIAL SETUP	QUEUE / CAP	SCORE	NOTES
● FL-400A Filler FL-400A	WC-FILL Filling Station	available	✓	<div style="width: 50%; background-color: #007bff; height: 10px;"></div> 1/2	1.0	queue depth: 1
○ FL-400B Filler FL-400B	WC-FILL Filling Station	busy	✓	<div style="width: 0%; background-color: #6c757d; height: 10px;"></div> 0/2	—	unavailable (state: busy)

Close

Orders

The Orders tab displays information about orders that can be filtered by status. For example:

MES Runtime — Shop Floor 🔔 Live

Dashboard WIP **Orders** Inventory Equipment Live Events

Production Orders

+ New ▶ Release ✓ Complete ✎ Edit 🗑 Close

Status: All ↻ Refresh

Order #	Product	Status	Priority	Ordered	Completed	Scrapped	ERP Ref	Created
⌵ SQA-ECB-20260519193830	Electronic Controller Board v1	in progress	1	1 EA	0 EA	0 EA	—	5/19/2026, 12:38:31 PM

+ Create Unit

UNITS (3)

Serial #	Status	Step	Report
SN-SQA-ECB-20260519193830-00003	queued	—	Report
SN-SQA-ECB-20260519193830-00002	queued	—	Report
SQA-ECB-SN-20260519193830	in process	Automated Optical Inspection	Report

An order can be created or edited here as well as released to production, completed or closed. Units or lots can also be created here. The Report link on a unit or lot displays the WIP History Report.

Inventory

The inventory tab allows the user to execute inventory transactions. For example, Receive in the Operations tab which creates material or adds to material to a lot:

MES Runtime — Shop Floor 🔔 Live

Dashboard WIP Orders **Inventory** Equipment Live Events

Inventory

Operations Balances Material Lots Transaction Log

Receive Move Consume Adjust

Receive Inventory

Material Lot:

To Location:

Quantity:

Reason:

Submit Receive

Other transactions are Move, Consume and Adjust.

The Balances tab allows the user to search for inventory units or lots by id or by location. For example:

Inventory

Operations **Balances** Material Lots Transaction Log

Current Inventory Balances Refresh

Search by lot number... EB-WH-SMT — SMT Component Warehouse X Clear 6 balances

LOT #	MATERIAL	LOCATION	ON HAND	AVAILABLE
LOT-CONF-2026A	RM-CONFORMAL — Conformal Coating	EB-WH-SMT	30000	30000
LOT-FLUX-2026A	RM-FLUX — Flux Solution	EB-WH-SMT	20000	20000
LOT-SLDR-2026A	RM-SOLDER-PST — Solder Paste Cartridge	EB-WH-SMT	50000	50000
LOT-THRU-2026A	RM-THRU-KIT — Through-Hole Component Kit	EB-WH-SMT	1500	1500
LOT-SMD-2026A	RM-SMD-KIT — SMD Component Kit	EB-WH-SMT	1500	1500
LOT-PCB-2026A	RM-PCB-BLANK — Bare PCB Board	EB-WH-SMT	2000	2000

The Material Lots tab allows the user to search for material lots by material type or status. For example:

Inventory

Operations Balances **Material Lots** Transaction Log

RM-VITC — Vitamin C Powder All Statuses Refresh + New Lot

LOT #	MATERIAL	ON HAND	STATUS	RECEIVED	EXPIRY	SUPPLIER	ACTIONS
VTC-2026-001	RM-VITC — Vitamin C Powder	24.9	available	2026-03-15	2027-03-15	NutriChem Inc.	Edit

An existing lot's properties can be edited, or a new lot can also be created here.

The Transaction Log tab show a history of material lot transactions. For example:

Inventory

Operations Balances Material Lots Transaction Log

Transaction Log Refresh

Search by lot number... All All Locations X Clear 56 transactions

TIME	TYPE	MATERIAL LOT	FROM	TO	QTY	REASON
5/19/2026, 12:36:57 PM	adjust	LOT-ESDB-2026A	EB-RIP-SMT	—	5	SQA Manual Adjustment
5/19/2026, 12:36:56 PM	consume	LOT-ESDB-2026A	EB-SHIP-01	—	15	—
5/19/2026, 12:36:55 PM	move	LOT-ESDB-2026A	EB-SHIP-01	EB-RIP-SMT	5	—
5/19/2026, 12:36:54 PM	receive	LOT-ESDB-2026A	—	EB-SHIP-01	100	—
5/16/2026, 3:08:35 PM	adjust	LOT-ESDB-2026A	EB-RIP-SMT	—	5	SQA Manual Adjustment
5/16/2026, 3:08:34 PM	consume	LOT-ESDB-2026A	EB-SHIP-01	—	15	—
5/16/2026, 3:08:33 PM	move	LOT-ESDB-2026A	EB-SHIP-01	EB-RIP-SMT	5	—
5/16/2026, 3:08:32 PM	receive	LOT-ESDB-2026A	—	EB-SHIP-01	100	—
5/16/2026, 12:27:10 PM	adjust	LOT-ESDB-2026A	EB-RIP-SMT	—	5	SQA Manual Adjustment
5/16/2026, 12:27:08 PM	consume	LOT-ESDB-2026A	EB-SHIP-01	—	15	—
5/16/2026, 12:27:07 PM	move	LOT-ESDB-2026A	EB-SHIP-01	EB-RIP-SMT	5	—
5/16/2026, 12:27:06 PM	receive	LOT-ESDB-2026A	—	EB-SHIP-01	100	—
5/15/2026, 4:14:46 PM	adjust	LOT-ESDB-2026A	EB-RIP-SMT	—	5	SQA Manual Adjustment

Equipment

The equipment tab allows the user to perform a material setup, and monitor and record equipment performance. For example:

MES Runtime — Shop Floor Live

Dashboard WIP Orders Inventory Equipment Live Events

Material Setup Monitor Performance

SELECT EQUIPMENT

- APEX-ELEC
- PCBA-AREA
 - LINE-SMT-01
 - WC-TEST
 - WC-THT
 - WC-AOI
 - AOI-300
 - WC-REFLOW
 - WC-PLACE
 - WC-PASTE
 - WC-REWORK
 - WC-PACK
 - SB-PLANT

CURRENT MATERIAL SETUP — AOI-300 (AOI CAMERA SYSTEM)

Material: **Electronic Controller Board v1** (FG-ECB-100) Job Number: **Job 2**
Design Speed: 180 Setup At: 5/21/2026, 4:05:02 PM

Clear Setup

SWITCH MATERIAL

Material: Job Number: Switch Material

On the material setup tab, material and/or job number can be set.

On the Monitor tab, the refresh rate can be set to update the display of equipment

properties. For example:

Refresh Refresh every seconds Monitoring

EQUIPMENT	DESCRIPTION	MATERIAL CODE	MATERIAL NAME	QUEUED	IN PROCESS	WIP
FCT-200	Functional Test Fixture	-	—	0	0	—
WS-100	Wave Solder + Coat Station	-	—	0	0	—
AOI-300	AOI Camera System	FG-ECB-100	Electronic Controller Board v1	0	1	units
RO-500	5-Zone Reflow Oven	-	—	0	0	—
PNP-800B	Pick-and-Place PNP-800B	-	—	0	0	—
PNP-800A	Pick-and-Place PNP-800A	-	—	0	0	—
SP-200	Stencil Printer SP-200	-	—	0	0	—
RW-BENCH	Rework Bench	-	—	0	0	—
RW-600	Adjustment Tank RW-600	-	—	0	0	—
PACK-100	ESD Packaging Station	-	—	0	0	—
LP-500	Labeler/Packer LP-500	-	—	0	0	—

seconds

Monitoring 11 equipment

STATE	IN STATE	DISPATCH	QUEUE CAP.	OEE BUCKET
Scheduled Downtime	28 min	unavailable_planned	1	downtime_planned
Unscheduled Downtime	29 min	unavailable_unplanned	1	downtime_unplanned
Aborting	29 min	busy	1	downtime_unplanned
Productive	29 min	busy	5	uptime_value_add
Aborting	29 min	busy	2	downtime_unplanned
Aborting	28 min	busy	2	downtime_unplanned
Aborting	29 min	busy	1	downtime_unplanned
Standby	—	available	3	uptime_non_value
Standby	—	available	2	uptime_non_value
Standby	—	available	10	uptime_non_value
Stopped	—	unavailable_planned	2	downtime_planned

Live Events

The Live Events tab shows events received on the event bus. They can be filtered by type. For example:

Live Events Clear

All WIP Orders Dispatch Data Equipment

```
6:52:19 PM equipment.state.changed performance
{"equipment_id":"46be4b24-274e-4a3d-a632-e95a11fa79c6","state":"Productive","dispatch_category":"busy"}

6:52:19 PM wip.lot.started wip
{"lot_id":"493db542-ba38-470d-996c-aa6635430eb1","step_id":"e769b2c2-7ee6-4265-9493-3f2840ee07d6","equipment_id":"46be4b24-274e-4a3d-a632-e95a11fa79c6"}

6:52:14 PM dispatch.executed dispatch
{"unit_id":null,"lot_id":"493db542-ba38-470d-996c-aa6635430eb1","destination_step_id":"e769b2c2-7ee6-4265-9493-3f2840ee07d6","destination_equipment_id":"46be4b24-274e-4a3d-a632-e95a11fa79c6"}

6:52:14 PM dispatch.evaluated dispatch
{"unit_id":null,"lot_id":"493db542-ba38-470d-996c-aa6635430eb1","strategy":"shortest_queue","recommendation":"46be4b24-274e-4a3d-a632-e95a11fa79c6"}

6:52:14 PM wip.lot.moved wip
{"lot_id":"493db542-ba38-470d-996c-aa6635430eb1","from_step_id":"32344c31-d266-4689-b827-cb49eb164faf","to_step_id":"e769b2c2-7ee6-4265-9493-3f2840ee07d6"}

6:52:14 PM wip.lot.completed wip
{"lot_id":"493db542-ba38-470d-996c-aa6635430eb1","step_id":"32344c31-d266-4689-b827-cb49eb164faf","quantity_out":200,"quantity_scrapped":0}
```

Showing 6 of 6 events (6 total received)

ERP Simulator (erp-sim)

erp-sim is a simulator used to create and release operations requests and exercise ERP-style inbound and outbound touchpoints. It is installed as an optional plugin for Oracle or SAP. Only one ERP plugin can be enabled at a time.

For example, here is the dashboard for Oracle:

The screenshot displays the Oracle ERP Simulator Dashboard. On the left is a dark navigation sidebar with the 'Ora ERP Simulator' logo and menu items categorized under 'INBOUND (ERP → MES)' (Dashboard, Production Orders, Materials, Products) and 'OUTBOUND (MES → ERP)' (Report Completion, Report Consumption, Report Scrap, Confirmations). The main content area is titled 'ERP Simulator Dashboard' and includes a 'Monitor adapter health and connectivity to the Oracle ERP Simulator plugin.' section. This section features an 'Adapter Health' overview with a 'Check Health' button and two status cards: 'Inbound Adapter' (Connected — ERP → MES sync ready, Health: Healthy) and 'Outbound Adapter' (Connected — MES → ERP reporting ready, Health: Healthy). Below this is a 'Quick Reference' section with two columns of links: 'Inbound (ERP → MES)' (Production Orders, Materials, Products, Bills of Material, Routings, Work Centers) and 'Outbound (MES → ERP)' (Production Completion, Material Consumption (WIP Issue), Scrap Report).

Inbound

Production Orders

The Production Orders link shows a page with active orders and the capability to create, edit and delete orders:

☰ Production Orders

Refresh Cancel 4 orders

Create Production Orders

Product * # Orders Qty per Order Priority

Order Number (suffixed -001, -002... if provided) ERP Reference (suffixed -001, -002...)

[Create 3 Orders](#)

ERP REF	ORDER #	STATUS	QTY ORDERED	QTY DONE	QTY SCRAP	PRIORITY	ACTIONS
ABC	OJ-052126	created	100	0	0	1	Edit Delete
1234	FG-OJ-1L-MPG9VTWD-001	created	100	0	0	1	Edit Delete
—	05212026-1	in_progress	200	0	0	0	Edit Delete
—	SQA-ECB-20260519193830	in_progress	1	0	0	1	Edit Delete

Materials

The Materials link displays materials and allows the user to edit or delete them. For example:

Materials

[Read Materials](#)[+ Add Material](#)

20 materials

CODE	NAME	ORACLE TYPE	UOM	REV	DESCRIPTION	SHELF LIFE	ACTIONS
FG-ECB-100	Electronic Controller Board v1	finished	EA	—		—	Edit Delete
PKG-ESD-BAG	ESD Protective Bag	packaging	EA	—		—	Edit Delete
SF-POP-PCB	Populated PCB Assembly	semi_finished	EA	—		—	Edit Delete
RM-CONFORMAL	Conformal Coating	raw	mL	—		—	Edit Delete
RM-FLUX	Flux Solution	raw	mL	—		—	Edit Delete
RM-SOLDER-PST	Solder Paste Cartridge	raw	g	—		—	Edit Delete
RM-THRU-KIT	Through-Hole Component Kit	raw	EA	—		—	Edit Delete
RM-SMD-KIT	SMD Component Kit	raw	EA	—		—	Edit Delete
RM-PCB-BLANK	Bare PCB Board	raw	EA	—		—	Edit Delete
FG-OJ-1L	Premium Orange Juice 1 L	finished	EA	—		—	Edit Delete
PKG-CASE	Shipping Case (12-pack)	packaging	EA	—		—	Edit Delete
PKG-LABEL	Product Label	packaging	EA	—		—	Edit Delete
PKG-CAP	Bottle Cap	packaging	EA	—		—	Edit Delete
PKG-BOTTLE-1L	PET Bottle 1 L	packaging	EA	—		—	Edit Delete
SF-JUICE-BLEND	Blended Juice	semi_finished	L	—		—	Edit Delete
RM-VITC	Vitamin C Powder	raw	kg	—		—	Edit Delete
RM-CITRIC	Citric Acid	raw	kg	—		—	Edit Delete
RM-SUGAR	Cane Sugar	raw	kg	—		—	Edit Delete
RM-WATER	Purified Water	raw	L	—		—	Edit Delete

Products

The Products link displays produced material along with its BOM and ERP routing. It allows the user to delete a product. For example:

Products

Read Products

Delete

#	CODE	NAME	TYPE	VERSION	UOM	DESCRIPTION
1	FG-ECB-100	Electronic Controller Board v1	discrete	1.0	EA	Series-100 electronic controller board with SMD and through-hole components
2	FG-OJ-1L	Premium Orange Juice 1 L	process	1.0	EA	1-litre PET bottle of premium orange juice

Bill of Materials

Routing

BOM v1.0

Pos	Material	Qty	UoM
80	PKG-ESD-BAG	1	EA
70	SF-POP-PCB	1	EA
60	RM-CONFORMAL	3	mL
50	RM-FLUX	2	mL
40	RM-SOLDER-PST	5	g
30	RM-THRU-KIT	1	EA
20	RM-SMD-KIT	1	EA
10	RM-PCB-BLANK	1	EA

Bill of Materials

Routing

SMT Assembly Line v1.0 (default)

Seq	Name	Type	Cycle Time (s)	ERP Op #
90	Final Packaging & Labeling	production	45	0090
80	MRB Review	mrB	600	0080
70	Rework Station	rework	300	0070
60	Functional Test	inspection	60	0060
50	Through-Hole & Conformal Coat	production	120	0050
40	Automated Optical Inspection	inspection	20	0040
30	Reflow Soldering	production	180	0030
20	SMD Placement	production	45	0020
10	Solder Paste Application	production	30	0010

Outbound

Report Completion

This dialog tests sending a step completion with good and rejected quantities. Fro example:

☰ **Report Completion**

Post a production completion confirmation to SAP (MB31 equivalent).

Production Order

SQA-ECB-20260519193830 — in_progress ▾

Qty Good **Qty Reject**

3 1

Operation Step

60. Functional Test (inspection) [Op 0060] ▾

Post Completion

Report Consumption

This dialog tests sending a consumption report with the order, BOM item material, material lot and quantity. For example:

☰ **Report Consumption**

Post a goods movement 261 (material consumption) to SAP.

Production Order

FG-OJ-1L-MPG9VTWD-001 (1234) — in_progress ▾

Add BOM Material

— pick a material — ▾ Add

Materials to Consume

Material	Qty	UoM	Lot #
RM-CITRIC	2	kg	CIT-2026-001 (99.6 kg) ▾ ×

Post Consumption (261)

Report Scrap

This dialog tests sending a scrap report with the order number, quantity scrapped and reason. For example:

☰ Report Scrap

Post a goods movement 531 (scrap posting) to SAP.

Production Order

 ▼

Qty Scrapped

Reason Code

Post Scrap (531)

Confirmations

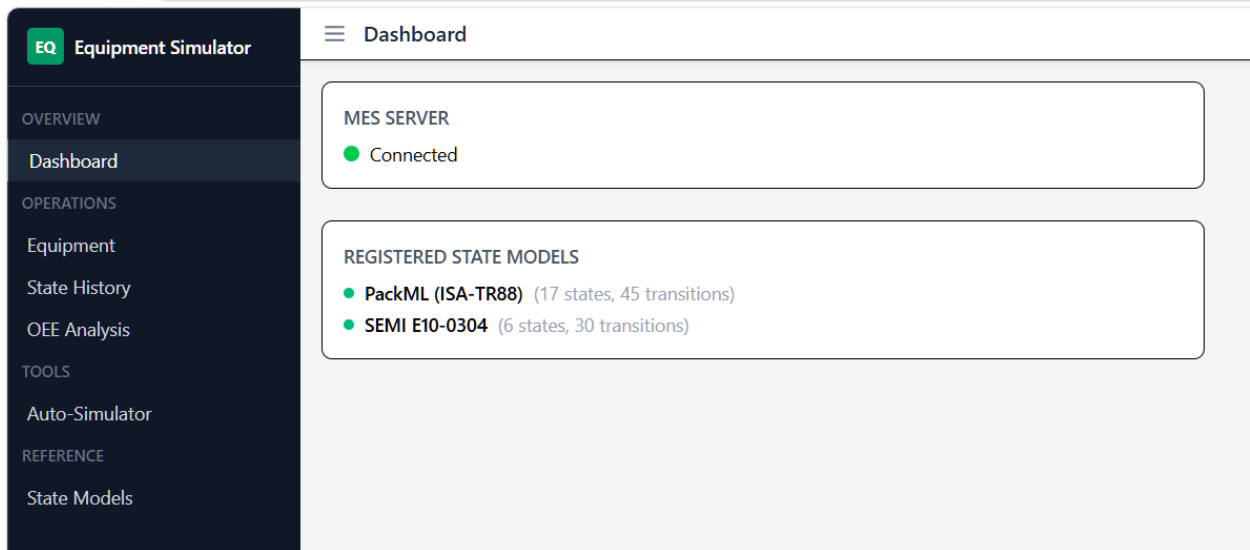
Successful reports to the ERP will be listed in this page.

Equipment Simulator (equipment-sim)

equipment-sim is a simulator used for equipment development and testing. It is installed as an optional plugin.

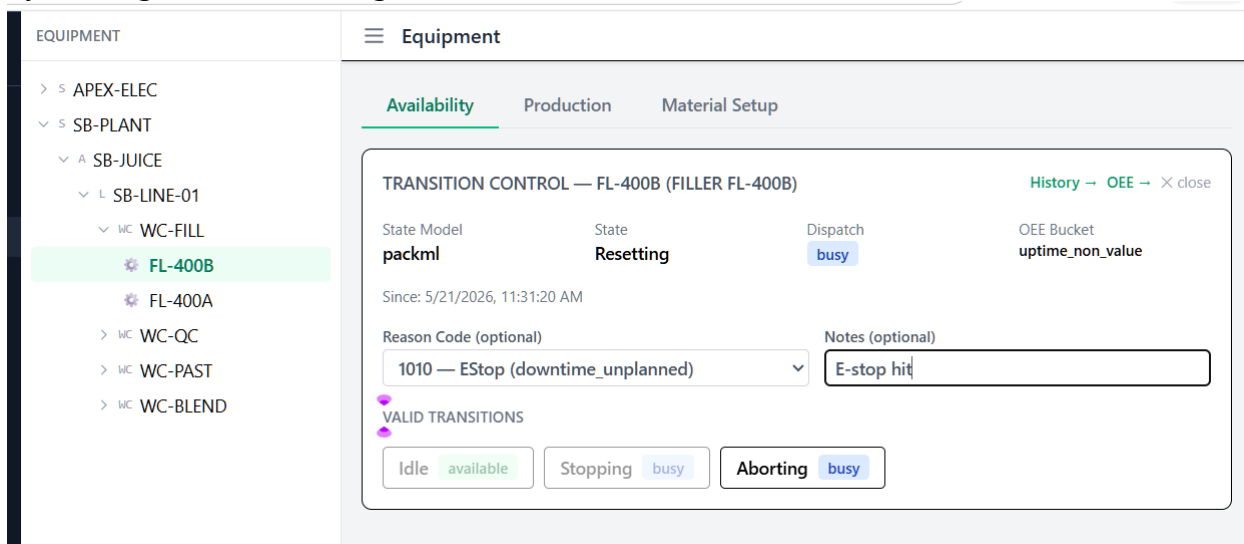
Dashboard

For example, PackML and SEMI E10 are installed state models:



Equipment

Availability, production and material setup events are entered on this page. For example filler FL-400B has an E-stop event and will be transitioned to the PackML Aborting state by clicking on the Aborting button:



and for production using the PackML counter tags:

EQUIPMENT

- > APEX-ELEC
- ▼ SB-PLANT
 - ▼ SB-JUICE
 - ▼ SB-LINE-01
 - ▼ WC-FILL
 - FL-400B**
 - FL-400A
 - ▼ WC-QC
 - ▼ WC-PAST
 - ▼ WC-BLEND

Equipment

Availability **Production** Material Setup

PRODUCTION COUNTS — FL-400B

Enter processed (good), defective (reject), and rework counts. Values are added as deltas to today's shift counter via the PackML Admin.ProdProcessedCount / Admin.ProdDefectiveCount model.

GOOD

10

REJECT

0

REWORK

1

Processed (Good) Defective (Reject) Rework

10 0 01 Submit Counts

Counts updated: +10 good, +1 rework — totals: 10 good, 0 reject, 1 rework

and for a material setup:

EQUIPMENT

- > APEX-ELEC
- ▼ SB-PLANT
 - ▼ SB-JUICE
 - ▼ SB-LINE-01
 - ▼ WC-FILL
 - FL-400B**
 - FL-400A
 - ▼ WC-QC
 - ▼ WC-PAST
 - ▼ WC-BLEND

Equipment

Availability Production **Material Setup**

CURRENT MATERIAL SETUP

Material: **Premium Orange Juice 1 L (FG-OJ-1L)** Job Number: **Job 456**

Design Speed: **300** Setup At: **5/23/2026, 11:37:31 AM**

Clear Setup

SWITCH MATERIAL

Material: Job Number: Switch Material

State History

This page displays the history of state transitions. For example:

State History

Showing history for **FL-400B** — Filler FL-400B

Max rows: 50 Refresh

STATE	MODEL	DISPATCH	OEE BUCKET	REASON	NOTES	STARTED	ENDED	DURATION
Aborting	packml	busy	downtime_unplanned	1010	E-stop hit	5/23/2026, 11:34:35 AM	active	3m 30s
Resetting	packml	busy	uptime_non_value	—	—	5/21/2026, 11:31:20 AM	5/23/2026, 11:34:35 AM	48h 3m

Showing 2 entries (newest first)

OEE Analysis

This page calculates Overall Equipment Effectiveness (OEE) for a specified time period. For example:

☰ OEE Analysis

Equipment: **FL-400B** — Filler FL-400B

TIME PERIOD

Start: 04/01/2026 12:00 AM

End: 05/23/2026 12:00 PM

OEE

1.4%

AVAILABILITY

99.1%

PERFORMANCE

1.5%

QUALITY

98.1%

DETAILS

planned production time sec	run time sec	excluded time sec
174519.79	172995.45	0.00
ideal cycle time sec	total good	total reject
12.00	210.00	1.00
total rework	total count	state log count
3.00	214.00	2.00

SIX BIG LOSSES

planned downtime sec	unplanned downtime sec	speed loss sec
0.00	1524.35	170427.45
quality loss count		
4.00		

Auto-Simulator

This page simulates equipment state changes using valid transitions according to the assigned state model. For example:

SIMULATION CONTROLS

Interval (seconds): Start Simulation Reload Equipment Clear Log

Equipment: **16 / 16** Transitions: **2** Errors: **0**

EQUIPMENT STATES (16)

<p>FCT-200 Functional Test Fixture State: Scheduled Downtime unavailable_planned downtime_planned</p>	<p>WS-100 Wave Solder + Coat Station State: Unscheduled Downtime unavailable_unplanned downtime_unplanned</p>	<p>AOI-300 AOI Camera System State: Aborting busy downtime_unplanned</p>	<p>RO-500 5-Zone Reflow Oven State: Engineering busy uptime_non_value → Engineering at 11:42:14 AM</p>
<p>PNP-800B Pick-and-Place PNP-800B State: Aborting busy downtime_unplanned</p>	<p>PNP-800A Pick-and-Place PNP-800A State: Aborting busy downtime_unplanned</p>	<p>SP-200 Stencil Printer SP-200 State: Aborting busy downtime_unplanned</p>	<p>RW-BENCH Rework Bench State: Standby available uptime_non_value</p>
<p>RW-600 Adjustment Tank RW-600 State: Standby available uptime_non_value</p>	<p>PACK-100 ESD Packaging Station State: Standby available uptime_non_value</p>	<p>LP-500 Labeler/Packer LP-500 State: Stopped unavailable_planned downtime_planned</p>	<p>FL-400B Filler FL-400B State: Aborting busy downtime_unplanned</p>
<p>FL-400A Filler FL-400A State: Resetting busy uptime_non_value → Resetting at 11:42:04 AM</p>	<p>QC-300 Lab Analyzer QC-300 State: Productive busy uptime_value_add</p>	<p>PS-200 HTST Pasteurizer PS-200 State: Stopping busy downtime_planned</p>	<p>TM-100 Tank Mixer TM-100 State: Non-Scheduled unavailable_planned excluded</p>

TRANSITION LOG (2)

TIME	EQUIPMENT	FROM	TO	DISPATCH	STATUS
11:42:15 AM	RO-500	Productive	Engineering	busy	OK
11:42:04 AM	FL-400A	Stopped	Resetting	busy	OK

State Models

This page is a reference to the installed state models, e.g. PackML and SEMI E10. For example:

PackML (ISA-TR88)

SEMI E10-0304

MODEL INFO

Model ID
packml

Initial State
Stopped

ISA-TR88 / PackML state model. 17 states organised around the Execute production cycle with Hold, Suspend, Stop and Abort branches.

STATES (17)

NAME	DISPATCH	OEE BUCKET
Stopped (initial)	unavailable_planned	downtime_planned
Idle	available	uptime_non_value
Complete	available	uptime_non_value
Held	unavailable_unplanned	downtime_unplanned
Suspended	unavailable_planned	downtime_planned
Aborted	unavailable_unplanned	downtime_unplanned
Resetting	busy	uptime_non_value
Starting	busy	uptime_value_add
Execute	busy	uptime_value_add
Completing	busy	uptime_value_add
Holding	busy	downtime_unplanned
Unholding	busy	downtime_unplanned
Suspending	busy	downtime_planned
Unsuspending	busy	downtime_planned
Stopping	busy	downtime_planned
Aborting	busy	downtime_unplanned
Clearing	busy	downtime_unplanned

TRANSITIONS (45)

FROM	TO	TRIGGER
Stopped	Resetting	—
Resetting	Idle	—
Idle	Starting	—
Starting	Execute	—
Execute	Completing	—
Completing	Complete	—
Complete	Resetting	—
Execute	Holding	—

Software Quality Assurance

REST API Unit Tests

All the unit tests can be run by executing pytest:

```
cd server
pytest tests/unit
```

There are scoping options which can be found by prompting the AI for them.

User Interface Tests

Automated user interface CRUD (Create, Read, Update and Delete) testing for design time client (dt-client) editors were implemented. In the SQA folder, the script is called run-dt-audit.ps1/sh. The scope argument defines the scope of the test, for example:

```
./run-rt-audit.ps1 -Scope all
```

Executing the script without any arguments prints a usage/help.

Documentation

REST API

The REST API is documented in three ways:

Swagger UI	<a href="http://<mes host:port>/api/v1/docs">http://<mes host:port>/api/v1/docs
ReDoc	<a href="http://<mes host:port>/api/v1/redoc">http://<mes host:port>/api/v1/redoc
OpenAPI JSON	<a href="http://<mes host:port>/api/v1/openapi.json">http://<mes host:port>/api/v1/openapi.json

Swagger UI <http://localhost:8082/api/v1/docs>
ReDoc <http://localhost:8082/api/v1/redoc>
OpenAPI JSON <http://localhost:8082/api/v1/openapi.json>

Architecture

The docs\ARCHITECTURE.md document contains the technical details of the system architecture. It was created and maintained by an AI agent. Before making changes with an AI agent, have the AI read this document and then keep it up-to-date as the project progresses.

As changes are made, have the AI maintain a PROJECT_STATE.json document that is created at the beginning of a project with detailed task instructions, then kept current as the project progresses. Along with this document, have the AI maintain a SESSION_LOG.md which preserves what happened in the current session. When a new session is started, instruct the AI to read both of these documents in order to continue where you left off.

Further Study

Using your preferred IDE and AI model, implement one or more of these items:

- Write an operator app in another technology stack that calls the REST API, for example a native iOS or Android application.
- Implement a Kafka messaging data source plugin
- Implement an equipment preventive maintenance plugin.
- Localize text, dates and times
- Implement import/expot of design time entities for transfer to another database
- Implement GitHub Continuous Integration (CI) workflow
- Implement database index and view creation scripts
- Make Vite apps Progressive Web Apps (PWAs)
- Implement clone/save as in design time editors
- Implement a realistic custom runtime dispatch algorithm
- Implement an order management workflow prior to order release
- Provide a realistic implementation of the file drop equipment interface
- Implement Non-Conformance Reporting (NCR) and resolution tracking
- Implement operator certification and training
- Implement a monitoring dashboard to show Key Performance Indicators (KPIs) such as Overall Equipment Effectiveness (OEE) and display notifications from the event bus.
- Implement local and OIDC authentication for rt-client, ERP and equipment simulators

- Implement client authorization access based on the authentication
- Enhance the runtime inventory page to handle units as well as lots.