

# Other Work Instructions

- Welding
  - Doc#
  - Doc#
- APS Instructions
  - Resource and Capabilities
  - Attributes
  - Routing Templates
  - Storage Areas

# Welding

Welding

# Doc#

## ☐☐ Document Title

**Author:**

**Date Created:**

**Last Updated:**

**Version:**

**Status:** Draft | Review | Approved

---

## ☐☐ Overview

Briefly describe the purpose of this document.

- What is this about?
  - Who is it for?
  - Why does it exist?
- 

## ☐☐ Objectives

- Objective 1
  - Objective 2
  - Objective 3
- 

## ☐☐ Scope

Included

- Item 1
- Item 2

## Excluded

- Item A
  - Item B
- 

# ☐ Architecture / Design

## High-Level Overview

[User] → [Frontend] → [API] → [Database]

Welding

# Doc#

## ☐☐ Document Title

**Author:**

**Date Created:**

**Last Updated:**

**Version:**

**Status:** Draft | Review | Approved

---

## ☐☐ Overview

Briefly describe the purpose of this document.

- What is this about?
  - Who is it for?
  - Why does it exist?
- 

## ☐☐ Objectives

- Objective 1
  - Objective 2
  - Objective 3
- 

## ☐☐ Scope

Included

- Item 1
- Item 2

## Excluded

- Item A
  - Item B
- 

# ☐ Architecture / Design

## High-Level Overview

[User] → [Frontend] → [API] → [Database]

# APS Instructions

# Resource and Capabilities

**Audience:** ERP users who are familiar with Action Target's system but are new to PlanetTogether.

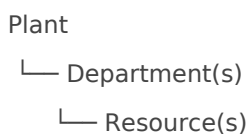
**Scope:** Phase 1 — resource master data, capabilities, and capability assignments. **Related**

**stories:** US-003 (Resource Maintenance Matrix), US-004 (Resource, Capability, and Attribute Maintenance Tools).

---

## 1. How PlanetTogether Organizes the Factory

Before you can understand resources and capabilities, it helps to understand how PlanetTogether structures the physical world. PT uses a three-level hierarchy:



### Plant

A **Plant** represents an entire facility or production location — in PlanetTogether terms, a single physical factory. For Action Target's Phase 1 integration, there is exactly **one plant**: Action Target Inc. (mapped from `base.ent.id = 1`).

The Plant is also the scope of a PT optimization run. When PT schedules jobs, it optimizes within the boundaries of a plant.

### Department

A **Department** is PlanetTogether's way of grouping resources that belong together. In most ERP systems, departments map closely to **work centers**. In Action Target's case, the four PT departments correspond to four vendor-organization rows in the ERP:

PT Department	ERP <code>vend_org.org_id</code>	ERP <code>org_name</code>
Fabrication	1196	AcTarg Fabrication
Assembly	1445	AcTarg Assembly
Electronics	1544	AcTarg Electronics
Wood Shop	1628	AcTarg Wood Shop

“ **What departments do in PT scheduling:** Departments have one scheduling effect — each department can have its own *frozen zone* (a window of time near the current clock where PT will not automatically move scheduled work). Otherwise, departments are organizational groupings, not scheduling constraints.

“ **Important limitation:** Once a resource is assigned to a department in PT, it cannot be reassigned. Similarly, a department cannot be reassigned to a different plant. If a reassignment is ever needed, the resource or department must be deleted and recreated under the correct parent. Keep this in mind when setting up the initial data.

Department setup and maintenance is managed by IT and is **read-only context** for the Planning Resources tools. You will see departments when setting up resources, but you will not edit them there.

## Resource

A **Resource** is the central scheduling object in PlanetTogether. A resource represents anything on the factory floor that performs work: a machine, a workstation, a labor pool, a tool, or any other entity that jobs must be scheduled on.

### Key things to understand about resources:

- Every resource belongs to one department (and therefore one plant).
- A resource must have at least one **capability** assigned to it before jobs can be scheduled on it (see Section 2).
- Each resource has a **Capacity Type** that controls how many operations it can run at once:
  - **SingleTasking** — Only one operation at a time. The operation must finish before the next one starts.
  - **MultiTasking** — A finite number of operations can run simultaneously, controlled by the number of people available on each shift interval.

- **Infinite** — Any number of operations may be scheduled simultaneously with no constraint. Used when a resource type truly has no practical limit (e.g., a holding area).
- Resources can also have **efficiency multipliers** for setup time and cycle time, allowing PT to model resources that run faster or slower than standard time.

### Action Target's Phase 1 resource pattern:

Resources are created at three granularities:

Resource type	One record per...	PT Resource Type
<b>Machine</b>	Individual machine on the floor	Machine
<b>Work Center</b>	Work center (grouping of related machines or stations)	WorkCenter
<b>Labor pool</b>	Pool of workers with a shared skill set	Labor

The current list of resources is maintained in `aps.resource_source` and managed through the Planning Resources screen. Do not rely on any static list in this document — the screen is always current.

## 2. Capabilities: Matching Resources to Work

### What a Capability Is

A **Capability** defines a *type of work* — a skill or process that a resource is qualified to perform. Think of it as a tag or classification. Examples might be "Welding," "Painting," "Assembly," or "CNC Machining."

Capabilities are the bridge between a job operation and the resources that can perform it. Here's how they connect:

1. A **capability** is created (e.g., "Welding").
2. That capability is **assigned** to one or more resources (e.g., the Weld resource and the PT Weld resource both receive the "Welding" capability).
3. A **job operation** declares that it *requires* the "Welding" capability.
4. PT's scheduler then knows which resources are eligible to schedule that operation on.



**In ERP terms:** Capabilities typically correspond to work centers in traditional ERP systems. But they are more flexible — a capability can also represent a special material type, a process constraint, or any other qualifying characteristic that limits which resources can do the work.

# Action Target's Phase 1 Capability Structure

Phase 1 uses three distinct capability types, each following a consistent naming convention that signals both intent and scheduling behavior at a glance:

Type	Naming pattern	What it means	Notes field value
<b>Labor</b>	Labor - [Skill]	Ties a labor resource to the type of work it performs. One labor capability per skill, assigned to exactly one labor resource.	(empty)
<b>Pool</b>	Pool - [Process]	Groups interchangeable resources. Any resource in the pool can satisfy the requirement.	Jobs can run on any resource in the pool.
<b>Machine</b>	Machine - [Machine Name]	Pins a requirement to a specific machine (or a very small set of equivalent machines).	Jobs can only run on specific resources.

The **Notes** field on a capability conveys intent to planners and reviewers. Pool capabilities say "Jobs can run on any resource in the pool," and Machine capabilities say "Jobs can only run on specific resources." Labor capabilities leave notes empty — the prefix makes the constraint self-evident.

## Why individual machines carry two capability types:

A machine like the Brake 400T is typically assigned *both* a Pool capability and a Machine-specific capability:

- The **Pool** capability (e.g., Pool - Bend) groups it with other interchangeable brakes so that generic bending operations can land on any eligible brake.
- The **Machine** capability (e.g., Machine - Bend 400T) allows operations that require *this specific press* — due to tonnage or tooling — to be pinned exclusively to it.

An operation routes to whichever capability matches its requirement: generic work links to the Pool, tonnage-specific work links to the Machine capability.

# Capability Assignments

A **Capability Assignment** is the record that links a specific capability to a specific resource. It requires:

- The capability being assigned
- The resource receiving it
- The plant and department context (derived automatically from the resource)

A resource can have many capabilities. A capability can be assigned to many resources. When a job operation requires *multiple* capabilities, a resource must have **all of the required capabilities** to be eligible — not just one of them.

“ **Example:** Suppose an operation requires both "Welding" and "Certified Inspection." Only a resource that has *both* capabilities assigned will be considered eligible to run that operation.

## Capability-Operation Links

The third piece of the capability picture is the **Capability-Operation Link**: a record that declares which capability a given ERP routing operation *requires*.

Think of it this way:

- **Resource-Capability Assignments** say: "*This resource can do this type of work.*"
- **Capability-Operation Links** say: "*This operation requires this type of work.*"

PT's scheduler matches them: an operation can only land on a resource when the resource has all the capabilities the operation's links require.

Each link also has a **Requirement Sequence** (Seq). The sequence number identifies which resource slot a capability requirement belongs to — PT must find a separate, simultaneously available resource to satisfy each distinct sequence number.

The sequence values recorded here are the **authoritative source** for which resource slots an operation uses. When jobs are exported to PT, one resource requirement row is created per distinct seq found in this table for the operation — not a fixed number. An operation that has only Seq 1 and Seq 3 links gets exactly two resource requirements. Among all of an operation's slots, the one with the **lowest seq** is marked as the primary requirement — the anchor slot PT treats as its scheduling driver. Action Target's Phase 1 design uses up to three slots per operation: the specific machine (Seq = 1), the pool the machine belongs to (Seq = 2), and a labor resource (Seq = 3).

# How Capabilities Affect Scheduling

PT will not schedule an operation on a resource unless that resource has all the capabilities the operation's links require. This means:

- Resources with no capabilities assigned cannot be scheduled on any operations.
- Operations with no capability link cannot be scheduled at all.
- Getting both sides right — what resources can do, and what operations require — is a prerequisite before job data can be exported to PT and scheduling can work correctly.

## 3. Our ERP Data Model: The `aps.*_source` Tables

All resource and capability data that flows to PlanetTogether is owned and maintained in the `aps` schema of the ERP database. PT is a *consumer* of this data — the ERP is the authoritative source of record.

The data lives in **source tables** (`aps.*_source`). PT-facing **export views** (`aps.*_v`) read from those source tables and present the data in the exact shape PT expects.

### Source Tables

Table	What it holds	PT object it feeds
<code>aps.plant_source</code>	The single Action Target plant record.	Plants
<code>aps.department_source</code>	The four production departments, linked to <code>public.vend_org</code> .	Departments
<code>aps.resource_source</code>	Resource master records; each belongs to a department/plant. Work area resources carry <code>workcenter</code> (display name) and <code>workcenter_id</code> (the associated <code>ati_pool</code> id from <code>work_order.ati_pools</code> , stored as varchar; no FK enforced).	Resources
<code>aps.capability_source</code>	Capability definitions (name, description, notes).	Capabilities
<code>aps.capability_assignment_source</code>	Which capabilities are assigned to which resources.	CapabilityAssignments

Table	What it holds	PT object it feeds
<code>aps.capability_operation_link</code>	Links ERP routing operations to capabilities, each with a <code>requirement_seq</code> . This is the authoritative source for which resource slots an operation requires. Only operations with active rows here will have resource requirements generated on export.	<code>JobResourceCapabilities</code>
<code>aps.workcell_department_link</code>	Legacy lookup: maps workcell names to departments. Used for context; not a PT export.	(Internal reference)

## Export Views

The export views are the data layer PT reads from. You do not need to interact with them directly, but understanding what they filter is useful:

View	Filters / shape
<code>aps.plants_v</code>	Active plants only. Columns: <code>external_id</code> , <code>name</code> , <code>description</code> .
<code>aps.departments_v</code>	Active departments joined to the active plant.
<code>aps.resources_v</code>	Active resources only. Includes default <code>batch_type = 'None'</code> , <code>capacity_type</code> , <code>resource_type</code> . Exposes <code>workcenter</code> and <code>workcenter_external_id</code> (sourced from <code>workcenter_id</code> on <code>aps.resource_source</code> ) for work area resources; null for machines and labor.
<code>aps.capabilities_v</code>	Active capabilities only.
<code>aps.capability_assignments_v</code>	Active assignments where both the capability and resource are also active.

## 4. How Data Flows to PlanetTogether

Once your source tables are populated, the data flows to PT through export views:

`aps.*_source` tables → `aps.*_v` export views → PlanetTogether import

PT reads from the export views — it never reads directly from the source tables. The views:

- Filter out inactive records automatically ( `where active = true` ).
- Present columns in the exact names and shapes PT expects.
- Join related tables to resolve foreign keys into the values PT needs (e.g., `capability_assignment_source` is joined to both the capability and resource tables so only fully active assignments export).

**The ERP is authoritative.** PT consumes what the ERP sends. For Phase 1, changes made inside PT to resource data are not written back to the ERP source tables.

## 5. Using the Planning Resources Screen

The **Planning Resources** tab is located inside the **Product Information** app. It is the central place to maintain all resource and capability data before it is exported to PlanetTogether.

Access to create, edit, and retire records requires the **Manufacturing Engineering, Product Data Manager**, or **IT Admin** role. Planning and scheduling leads have view-only access.

The screen contains five data grids. Three of them interact with each other dynamically based on what row you select, which is explained at the end of this section.

### Main Toolbar

At the top of the screen, three toggle buttons control which grids are visible:

Button	What it does
<b>Hide / Show Resources</b>	Collapses or expands the Resources grid. Hiding the grid also clears any active resource selection, which resets the Resource Capabilities filter back to showing all assignments.
<b>Hide / Show Capabilities</b>	Collapses or expands the Capabilities grid. Hiding it clears any active capability selection, resetting both the Resource Capabilities filter and the Operation Links filter.
<b>Hide / Show Operation Links</b>	Collapses or expands the Capability-Operation Links grid. The other grids are unaffected.

# Readiness Issues

This grid runs a live validation check every time you open the screen or save a change. It shows problems that would prevent a successful export to PlanetTogether.

## Columns displayed:

Column	What it shows
Severity	How serious the issue is (e.g., Error, Warning).
Code	A short code identifying the type of issue.
Message	A plain-language description of the problem.
Resource	The Resource ID involved, if applicable.
Capability	The Capability ID involved, if applicable.

This grid is read-only. You resolve issues by correcting the underlying data in the Resources, Capabilities, Resource Capabilities, or Operation Links grids below, then the Readiness Issues grid will refresh automatically.

# Resources Grid

This grid shows all APS resources from `aps.resource_source`. It is the starting point for managing everything the factory floor can do.

**Columns displayed by default:** Resource ID, Resource Name, Plant, Department, Resource Type, Capacity Type, Active. (A Notes column is also available but hidden by default.)

## Toolbar actions:

Action	Behavior
<b>Add</b>	Opens the resource form in create mode. All required fields must be filled before saving.
<b>Edit</b>	Opens the resource form loaded with the selected row's data. Select exactly one row first.
<b>Retire</b>	Prompts for confirmation, then marks the resource inactive ( <code>is_active = false</code> ). The resource is never physically deleted. It will be excluded from PT exports and cannot be scheduled on new jobs, but any historical references remain intact.
<b>Show Inactive / Hide Inactive</b>	Toggles whether inactive (retired) resources appear in the list. Defaults to hiding inactive records. Toggling this clears the current resource selection.

Action	Behavior
<b>Show Only Unmapped / Show All Resources</b>	Filters the list to resources that have <b>no capability assignments at all</b> . These resources cannot be scheduled by PT. Use this to quickly identify gaps before exporting. Toggling this also clears the current selection.

### Resource form fields:

Field	Required	Notes
Resource Name	<input type="checkbox"/>	The display name in PT. Use properly-cased names ("Powder Coat," not "powder coat").
Plant	<input type="checkbox"/>	Defaults to the existing plant. Phase 1 always uses plant <input type="text" value="1"/> .
Department	<input type="checkbox"/>	Must match one of the four active departments. Pick from the dropdown.
Resource Type	<input type="checkbox"/>	Loaded live from the database schema. Choose <input type="text" value="Machine"/> for individual machines, <input type="text" value="WorkCenter"/> for work centers, or <input type="text" value="Labor"/> for labor pools.
Capacity Type	<input type="checkbox"/>	Loaded live from the database schema. Defaults to <input type="text" value="Infinite"/> for Phase 1. See Section 1 for what each value means in PT.
Notes		Optional free-text for internal use. Not sent to PT.

“ **Resource Type and Capacity Type** dropdown values are loaded directly from the database schema's CHECK constraint via the API. This means the options always reflect what the schema allows — no UI change is needed if the constraint is ever updated.

**Selecting a resource row:** Clicking a row in this grid selects that resource and filters the Resource Capabilities grid to show only that resource's capability assignments. Any capability selection is cleared at the same time.

## Capabilities Grid

This grid shows all APS capability definitions from .

**Columns displayed by default:** Capability ID, Name, Description, Active. (A Notes column is also available but hidden by default.)

**Toolbar actions:**

Action	Behavior
<b>Add</b>	Opens the capability form in create mode.
<b>Edit</b>	Opens the capability form for the selected row.
<b>Retire</b>	Prompts for confirmation, then marks the capability inactive. Never physically deleted.
<b>Show Inactive / Hide Inactive</b>	Toggles visibility of inactive capabilities. Defaults to hiding them. Clearing this also clears the current capability selection.
<b>Show Only Unmapped / Show All Capabilities</b>	Filters to capabilities that are not assigned to any resource. A capability with no resource assignments cannot be used to schedule anything. Use this to find capabilities that need to be assigned before exporting.

**Capability form fields:**

Field	Required	Notes
Capability ID	Read-only	Shown on the edit form for reference. On a new capability, the ID is generated automatically by the system — you do not supply it.
Name	<input type="text"/>	The display name in PT. This is what planners see when reviewing job requirements in PlanetTogether.
Description	<input type="text"/>	Optional. Describe the type of work this capability covers, which resources it applies to, or any constraints.
Notes	<input type="text"/>	Optional internal notes.

“ **Capability IDs are system-generated** when you add a new capability. You do not choose the ID. The generated ID flows to PT as the `Capabilities.ExternalId` and will be referenced by resource-capability assignments and operation links — so once created, a capability ID is permanent. Retire and recreate if a significant change is needed.

**Selecting a capability row:** Clicking a row selects that capability and simultaneously filters two grids:

1. **Resource Capabilities** — shows only assignments that include this capability (i.e., which resources can do this type of work).
2. **Capability-Operation Links** — shows only operation links for this capability (i.e., which operations require this type of work).

Any resource selection is cleared at the same time.

## Resource Capabilities Grid

This grid shows resource-to-capability assignments from `aps.capability_assignment_source`. It is the link that tells PT which resources are eligible to perform which types of work.

**Columns displayed by default:** Resource, Capability, Plant, Department, Active. (*Notes is available but hidden.*)

### Toolbar actions:

Action	Behavior
<b>Add</b>	Opens the assignment form. Choose the resource and capability; Plant and Department are filled in automatically from the selected resource.
<b>Edit</b>	Opens the assignment form for the selected row. The Resource and Capability fields are <b>locked</b> because they form the composite primary key and cannot be changed after creation. Only <b>Notes</b> can be edited.
<b>Retire</b>	Prompts for confirmation, then removes or inactivates the assignment.

### Assignment form fields:

Field	Add	Edit	Notes
Resource	<input type="checkbox"/> editable	<input type="checkbox"/> locked	Dropdown shows only active resources. When a resource is selected, Plant and Department are filled in automatically.
Capability	<input type="checkbox"/> editable	<input type="checkbox"/> locked	Dropdown shows only active capabilities.
Notes	editable	editable	Optional internal notes.

### How the filter works:

- If a **resource is selected** in the Resources grid: shows only that resource's assignments (i.e., what capabilities does this resource have?).

- If a **capability is selected** in the Capabilities grid: shows only assignments for that capability (i.e., which resources can do this work?).
- If **nothing is selected**: shows all assignments.

This makes the Resource Capabilities grid a two-direction lookup depending on what you click first.

## Capability-Operation Links Grid

This grid links ERP routing operations to the capabilities they require. It lives in a separate source table and is the mechanism that tells PT *what kind of work* each operation step needs, so PT can find eligible resources.

Every routing operation that will be scheduled in PT must have at least one link here. Without it, PT cannot find a resource for the operation and it will not schedule.

**Columns displayed by default:** Operation, Capability, Capability Name, Seq (Requirement Sequence), Active. (*Operation ID and Notes are also available.*)

### Toolbar actions:

Action	Behavior
<b>Add</b>	Opens the operation link form. Choose the operation and capability. If a capability is currently selected in the Capabilities grid, the form pre-fills that capability.
<b>Edit</b>	Opens the form for the selected row. The Operation and Capability fields are <b>locked</b> (they identify the link). Only <b>Requirement Sequence</b> and <b>Notes</b> are editable.
<b>Retire</b>	Prompts for confirmation, then retires the link.

### Operation link form fields:

Field	Add	Edit	Notes
Operation	<input type="checkbox"/> editable	<input type="checkbox"/> locked	Dropdown populated from active ERP routing operations.
Capability	<input type="checkbox"/> editable	<input type="checkbox"/> locked	Dropdown shows active capabilities. Pre-filled if a capability is selected.
Requirement Sequence	<input type="checkbox"/> editable	editable	A positive integer (minimum 1). Determines the resource requirement slot in PT — see explanation below.

Field	Add	Edit	Notes
Notes	editable	editable	Optional internal notes.

### Understanding Requirement Sequence:

The Requirement Sequence (Seq) identifies which resource slot a capability requirement belongs to. PT must find a separate, simultaneously available resource to fill each slot. Action Target's Phase 1 design uses up to three slots per operation:

Seq	Capability type	What PT finds
1	Machine (Machine - ...)	The specific machine (or machine class) the operation runs on. Drives the operation's run time and scheduling window.
2	Pool (Pool - ...)	A resource from the pool the machine belongs to. Constrains eligibility to machines within the defined group.
3	Labor (Labor - ...)	The labor resource required to run the operation. Must be available at the same time as the machine.

Not every operation has all three slots — for example, operations without a machine-specific requirement omit Seq = 1, and blast has only Seq = 1 (no pool or labor link in Phase 1).

### How the filter works:

- If a **capability is selected** in the Capabilities grid: shows only operation links for that capability (i.e., which operations require this capability).
- If **nothing is selected** (or a resource is selected): shows all operation links.

## How the Tables Work Together

The three middle grids — Resources, Capabilities, and Resource Capabilities — plus the Operation Links grid are all connected through row selections. Here is a summary of what happens when you click in each grid:

You click...	Resource Capabilities shows...	Operation Links shows...
A row in <b>Resources</b>	Only that resource's capability assignments	All links (unchanged)
A row in <b>Capabilities</b>	Only assignments for that capability	Only links for that capability
Empty space / deselect	All assignments	All links

Clicking in Resources clears any capability selection, and clicking in Capabilities clears any resource selection. The two grids are mutually exclusive — you are always viewing the assignments from one direction at a time.

This design lets you quickly answer two common questions:

- "What capabilities does the Weld resource have?" → Click Weld in the Resources grid.
- "Which resources can perform Welding, and which operations require it?" → Click Welding in the Capabilities grid.

## 6. Phase 1 Scope: What's Used and What's Not

Some PlanetTogether resource features are available in the software but are not being configured for Phase 1. This is intentional — start simple and add complexity as needed.

PT Feature	Phase 1 Status	Notes
Plants	<input type="checkbox"/> Used	Single plant (Action Target).
Departments	<input type="checkbox"/> Used	Four departments from ERP <code>vend_org</code> . Read-only in Phase 1 tools.
Resources	<input type="checkbox"/> Used	Maintained in <code>aps.resource_source</code> . One per machine, one per work center, one per labor pool.
Capabilities	<input type="checkbox"/> Used	Maintained in <code>aps.capability_source</code> .
Capability Assignments	<input type="checkbox"/> Used	Maintained in <code>aps.capability_assignment_source</code> .
Capability-Operation Links	<input type="checkbox"/> Used	Maintained in the operation capability link table. Required for job export.
Capacity Intervals / Calendars	<input type="checkbox"/> Not used	PT will use default shift definitions; no ERP-driven calendar import in Phase 1.
Helper Resources (Seq > 1)	<input type="checkbox"/> Used	Each operation links to up to three capability slots: Seq 1 = Machine, Seq 2 = Pool, Seq 3 = Labor. All three resources must be simultaneously available for PT to schedule the operation.
Cells	<input type="checkbox"/> Not used	Cell scheduling not needed initially.
Product Rules	<input type="checkbox"/> Not used	Confirmed not used for Phase 1.

PT Feature	Phase 1 Status	Notes
Compatibility Code Tables	<input type="checkbox"/> Not used	Not needed for Phase 1.
Cleanout Trigger Tables	<input type="checkbox"/> Not used	Not needed for Phase 1.

---

## 7. Summary: The Minimum You Need Before Jobs Can Schedule

Before PlanetTogether can schedule a job, the following must be in place:

- Plant record exists ( `aps.plant_source` ).
- Department records exist ( `aps.department_source` ) for all departments referenced by resources.
- Resource records exist and are active ( `aps.resource_source` ).
- Capability records exist and are active ( `aps.capability_source` ).
- Every resource that will carry job operations has at least one capability assigned ( `aps.capability_assignment_source` ).
- Every routing operation that will be scheduled in PT has at least one Capability-Operation Link pointing to a capability that is assigned to at least one active resource.

Missing any of these means PT either cannot find an eligible resource for an operation, or the operation will fail to schedule.

# Attributes

**Audience:** ERP users who are familiar with Action Target's system but are new to PlanetTogether.

**Scope:** Phase 1 — attribute definitions, attribute codes, and operation attribute links. **Related**

**stories:** US-004 (Resource, Capability, and Attribute Maintenance Tools).

---

## 1. What Attributes Do in PlanetTogether

Attributes are custom data properties that can be attached to job operations. They give the scheduler a way to describe *what kind of work* an operation is — beyond which machine and labor pool it needs — so PlanetTogether can group, sequence, and schedule setup or cleanout time intelligently.

Phase 1 uses attributes for two purposes:

**Sequence optimization (batching):** When PT builds a schedule, it can minimize the number of setup changeovers by grouping consecutive operations that share the same attribute value on the same resource. An attribute with `UseInSequencing = true` becomes available to the PT Sequence Planning board's optimize factors. Schedulers can weight each attribute and configure how many optimization points to award for grouping similar work together.

**Setup and cleanout time:** When an attribute value changes between two consecutive operations on the same resource, PT can automatically schedule a setup or cleanout block between them. The `AttributeTrigger` property controls *when* the timer fires (e.g., always, when the code changes, when a numeric value rises), and `AttributeType` controls whether the time is counted as Setup or Clean time.

An attribute can serve both purposes at the same time, or just one, or purely as informational metadata visualized in the Gantt and scheduling grids.

---

# 2. The Three Layers: Attributes, Codes, and Operation Links

Working with attributes in the ERP involves three layers that build on each other:

## Layer 1 — Attribute Definition

An **Attribute** is the property type itself — a named, configurable object that describes a characteristic of work. Examples: "Steel Grade," "Coating Type," "Product Family." The attribute definition sets all the defaults: how PT should visualize it, whether it drives sequencing or setup calculations, and what duration or cost to use when it fires.

## Layer 2 — Attribute Codes

An **Attribute Code** is one allowed value for an attribute. Each attribute has a list of codes representing the discrete values it can take. For example, the "Steel Grade" attribute might have codes `A36`, `A500-B`, `AR200`, and `HRPO`.

Codes are the strings that get attached to individual operations. When two consecutive operations on the same resource carry *different* codes for the same attribute, PT can recognize the code change and trigger a setup or sequencing decision.

## Layer 3 — Operation Attribute Links

An **Operation Attribute Link** is the record that ties a specific code to a specific routing operation. It says: "When this operation runs, it should carry this attribute value."

Links live on the router — both the workbench (draft) and production sides. When a workbench router is published, its attribute links are copied forward to the production router. When a template is generated from the production router, the attribute links become rows in `aps.job_operation_attribute_source`, which PT reads for scheduling.

---

# 3. The `aps.*_source` Tables

Table	What it holds	PT object it feeds
-------	---------------	--------------------

<code>aps.attribute_source</code>	Attribute definitions (one row per attribute).	Attributes
<code>aps.attribute_code_source</code>	Allowed code values per attribute.	Referenced by Operation Attributes
<code>aps.bom_operations_attributes</code>	Operation attribute/code links for workbench and production routers.	(join source for template generation)
<code>aps.job_operation_attribute_source</code>	Generated per-operation attribute rows for template MOs.	OperationAttributes

`aps.attribute_code_table_source` exists in the schema but is not used in Phase 1.

## 4. How Data Flows to PlanetTogether

```

aps.attribute_source      → PT Attributes import
aps.attribute_code_source → (read at template generation time)
aps.bom_operations_attributes → (read at template generation time)
    ↓ GenerateRouterTemplate
aps.job_operation_attribute_source → PT Operation Attributes import

```

Attribute definitions are imported into PT as Attribute objects. The code values assigned to specific operations are carried in the template's Operation Attributes rows, generated each time a template is created or regenerated for a router. PT reads both when building a scenario.

**The ERP is authoritative.** Changes made inside PT to attribute data are not written back to the ERP source tables.

## 5. Using the Planning Attributes Screen

The **Planning Attributes** tab is located inside the **Product Information** app. It is where you define attributes and manage their allowed code values before linking them to router operations.

Access to create, edit, and retire records requires the **Manufacturing Engineering, Product Data Manager**, or **IT Admin** role. Planning and scheduling leads have view-only access.

The screen contains two side-by-side grids. The Codes grid is filtered by whatever attribute is selected in the Attributes grid.

## Main Toolbar

Button	What it does
Hide / Show Attributes	Collapses or expands the Attributes grid.
Hide / Show Codes	Collapses or expands the Codes grid.
Show Inactive / Hide Inactive	Toggles visibility of inactive attributes. Defaults to hiding them.

## Attributes Grid

This grid shows all APS attribute definitions from `aps.attribute_source`.

**Columns displayed by default:** Attribute ID, Name, Description, Use in Sequencing, Active.

### Toolbar actions:

Action	Behavior
Add	Opens the attribute form in create mode.
Edit	Opens the attribute form loaded with the selected row's data. Select exactly one row first.
Retire	Prompts for confirmation, then marks the attribute inactive. Never physically deleted.

### Attribute form fields:

#### Identity

Field	Required	Notes
Attribute ID	<input type="checkbox"/>	Must match the ID used in PT imports. Convention: <code>ATTR-[ShortName]</code> (e.g., <code>ATTR-SteelGrade</code> ). Locked after creation — retire and recreate if a change is needed.
Name	<input type="checkbox"/>	Display name in PT and in ERP grids.
Description		Describe what characteristic this attribute classifies.

## PT Setup Behavior

Field	Notes
Trigger	Controls when a setup or cleanout fires. See the trigger table below. Leave blank when using the attribute for sequencing only.
Type	<code>Setup</code> or <code>Clean</code> . Defines whether the incurred time counts as setup or cleanout. Leave blank for informational or sequencing-only attributes.
Cleanout Grade	Integer priority for simultaneous cleanout scheduling. When multiple cleanouts fire at the same time, the highest grade wins. A cleanout also resets all triggers at or below its own grade. Only relevant when Type = <code>Clean</code> .
Color Code	Hex color ( <code>#RRGGBB</code> ) for Gantt attribute segments.

### AttributeTrigger accepted values:

Trigger value	When the setup or clean fires
<code>Always</code>	Every time this operation is scheduled, regardless of what comes before it.
<code>CodeChanges</code>	When this attribute's code changes from one scheduled operation to the next on the same resource.
<code>NumberChanges</code>	When this attribute's number value changes from the previous operation to the next on the same resource.
<code>NumberHigher</code>	When this attribute's number value increases from the previous operation to the next.
<code>NumberLower</code>	When this attribute's number value decreases from the previous operation to the next.
<code>LookupByCode</code>	Duration is determined by an Attribute Code Table (Phase 2 only).
<code>LookupByRange</code>	Duration is determined by an Attribute Range Table (Setup only; Phase 2 only).
<code>Never</code>	This attribute is not used for setup or clean calculations.

### Defaults

Field	Notes
Default Cost	Cost incurred per setup or cleanout event triggered by this attribute. Not used with lookup table triggers.
Default Duration (hrs)	Hours incurred per event. Not used with lookup table triggers.

### Flags

Flag	Default	What it controls in PT
Consecutive Setup	false	When true, multiple setup steps for the operation must run sequentially rather than concurrently. Total setup time = max non-consecutive time + sum of all consecutive times.
Incur Resource Setup	false	When true, the sequenced setup fires even if the resource's <code>UseSequencedSetup</code> flag is false.
Show in Gantt	false	When true, this attribute's name and code appear in the Gantt activity block when the Attribute Segment is enabled on the label.
Hide in Grids	false	When true, this attribute's columns are hidden from the Activities board scheduling grids.
Use in Sequencing	false	When true, this attribute is available to the Sequence Planning board's optimize factors for schedule batching and grouping.

**Selecting an attribute row:** Clicking a row selects that attribute and filters the Codes grid to show only that attribute's allowed code values.

## Codes Grid

This grid shows the allowed code values for the currently selected attribute, from `aps.attribute_code_source`. When no attribute is selected, this grid is empty — select an attribute row first.

**Columns displayed by default:** Code, Display Name, Description, Active.

**Toolbar actions:**

Action	Behavior
<b>Add</b>	Opens the code form. The Attribute field is pre-filled and read-only. Disabled when no attribute is selected.
<b>Edit</b>	Opens the code form for the selected row.
<b>Retire</b>	Prompts for confirmation, then retires the code. Existing operation attribute links referencing a retired code remain historically valid — they are not automatically removed.
<b>Show Inactive / Hide Inactive</b>	Toggles visibility of inactive codes.

## Code form fields:

Field	Required	Notes
Attribute	Read-only	Pre-filled from the currently selected attribute.
Code	□	The short code string (e.g., <code>A36</code> , <code>COAT-POWDER</code> ). Must be unique within the attribute. This is the value PT carries on job operations.
Display Name		A longer, readable label shown in ERP dropdowns and grids.
Description		Optional explanation of what this code represents.
Notes		Optional internal notes.

“ **Code uniqueness:** Codes are unique within an attribute. The same code string can appear under two different attributes without conflict.

# 6. Linking Attributes to Operations (Routers Tab)

Attribute definitions and codes are only useful once they are attached to specific routing operations. This is done in the **Routers** tab, not the Attributes tab.

## Where to Find the Operation Attribute Links

In the Routers tab, select a workbench or production router and then select a BOM operation row. The **Operation Attributes** sub-grid below the operations grid shows the attribute/code links for that operation.

## Workbench vs. Production

Attribute links follow the same workbench/production pattern as the router itself:

Side	What it holds	When it updates
<b>Workbench</b>	Draft attribute links configured before publishing.	Add/edit/retire any time the router is being edited.
<b>Production</b>	Published links that feed template generation.	Replaced automatically when the workbench router is published.

Do not edit production operation attribute links directly for normal maintenance. Make changes on the workbench side, then publish.

## Operation Attribute Links Grid

**Columns displayed by default:** Attribute, Code, Active.

**Toolbar actions:**

Action	Behavior
<b>Add</b>	Opens the link form. Disabled when no BOM operation is selected or the router is not in an editable state.
<b>Edit</b>	Opens the link form for the selected row.
<b>Retire</b>	Prompts for confirmation, then retires the link. The operation will no longer carry this attribute when the next template is generated.

**Link form fields:**

Field	Required	Notes
Attribute	<input type="checkbox"/>	Dropdown showing active attributes.
Code	<input type="checkbox"/>	Dropdown showing active codes for the selected attribute.

“ **Multiple codes per operation:** An operation can have more than one attribute link, and can have multiple links for the *same* attribute with different codes. All active links are exported as separate rows in `aps.job_operation_attribute_source` .

## How Publishing Propagates Attribute Links

When a workbench router is published:

1. All existing production operation attribute links for that router are cleared.

2. The workbench links are copied to the production side, translated through the workbench-to-production BOM operation ID map.
3. The production links are now the authoritative source for the next template generation.

After publishing, regenerate the template so PT receives the updated attribute rows.

## 7. Phase 1 Scope: What's Used and What's Not

PT Feature	Phase 1 Status	Notes
Attribute definitions	<input type="checkbox"/> Used	Maintained in <code>aps.attribute_source</code> .
Attribute codes (ERP-owned allowed values)	<input type="checkbox"/> Used	Maintained in <code>aps.attribute_code_source</code> .
Operation attribute links	<input type="checkbox"/> Used	Configured per BOM operation in the Routers tab; propagated on publish; generated into templates.
UseInSequencing	<input type="checkbox"/> Used	Attributes with this flag true are eligible for sequence optimization in PT.
AttributeType / AttributeTrigger (Setup or Clean)	<input type="checkbox"/> Supported	Can be configured; actual scheduling of setup/clean time also requires the resource to have <code>SetupIncluded = UseOperationAttributes</code> (for Setup) or <code>UseAttributeCleanouts = true</code> (for Clean).
ShowInGantt / HideInGrids / ColorCode	<input type="checkbox"/> Supported	PT will respect these flags once attribute data is imported.
Attribute Code Tables ( <code>LookupByCode</code> trigger)	<input type="checkbox"/> Not used	Code Table data is not exported in Phase 1.
Attribute Range Tables ( <code>LookupByRange</code> trigger)	<input type="checkbox"/> Not used	Range Table data is not exported in Phase 1.
Number-based attribute values on operations	<input type="checkbox"/> Not used	Phase 1 operation links carry a code (string) value only. Triggers that depend on a numeric value ( <code>NumberChanges</code> , <code>NumberHigher</code> , <code>NumberLower</code> ) have no effect.

# 8. Summary: The Minimum You Need Before Attribute Data Can Influence Scheduling

For PT to use an attribute during scheduling:

1.  An attribute definition exists and is active ( `aps.attribute_source` ).
2.  At least one code is defined and active for that attribute ( `aps.attribute_code_source` ).
3.  At least one BOM operation on the production router has an active link to that attribute and code ( `aps.bom_operations_attributes` , `is_workbench = false` ).
4.  The template has been generated (or regenerated) after the production links were set — attribute rows must exist in `aps.job_operation_attribute_source` .
5.  **For sequence optimization:** `UseInSequencing = true` on the attribute, and the attribute must be selected in the applicable optimize factor on the PT Sequence Planning board.
6.  **For setup time:** `AttributeType = Setup` , a trigger configured, a default duration set, and the resource must have `SetupIncluded = UseOperationAttributes` .
7.  **For cleanout time:** `AttributeType = Clean` , a trigger configured, a default duration set, and the resource must have `UseAttributeCleanouts = true` .

Missing any of these means the attribute is defined but has no scheduling effect in PT.

# Routing Templates

**Audience:** ERP users who are familiar with Action Target's system but are new to PlanetTogether.

**Scope:** Phase 1 — router-to-template linking and template generation. **Related stories:** US-014 (Routing Maintenance Tools), US-015 (Router Template Generation).

---

## 1. What Is a Template?

PlanetTogether schedules jobs by matching job operations to eligible resources. To do that efficiently, PT uses **templates** — pre-built representations of how a part is manufactured — that it holds in memory and applies each time a new job for that part arrives.

In Action Target's integration, a template is derived directly from an ERP **router**. The router already defines the sequence of operations, setup and run times, and the materials consumed at each step. Template generation reads that router and produces a standardized PT job structure from it — without touching native work order tables.

### Key things to understand:

- Every part that will be scheduled in PT needs a template based on its current router.
  - A template is tied to one specific router revision. When a router is superseded by a new revision, that new revision needs its own template.
  - Generating a template is **safe to repeat** — re-running it refreshes the PT data from the current router state without creating duplicates.
  - Templates do not affect live work orders. They are scheduling reference structures used by PT, not production documents.
- 

## 2. Routers and the Concept of "Current"

An ERP **router** defines the manufacturing process for a part: the sequence of operations, the time each step takes, and the materials consumed. Routers are version-controlled — when a process changes, a new router revision is published for that part.

At any point in time, exactly one router revision is **current** for each part. A router is current when today's date falls within its effective date range ( **Effective From** through **Effective To** ). Older revisions are archived but remain visible in the system.

### Why this matters for templates:

- Only a current router can have a template generated from it. The system blocks generation on archived routers.
- When a router is superseded by a new revision, the old template is not automatically updated. The new revision must be selected in the Templates screen and a new template generated from it.
- Future automation of this step is planned but not in scope for Phase 1.

## 3. What Gets Generated

When you generate a template from a router, the system produces a complete PT job structure covering:

Generated object	What it represents
<b>Template Job</b>	The top-level PT job anchor for this router/part.
<b>Manufacturing Order</b>	One synthetic MO for the template — no sub-assembly expansion.
<b>Operations</b>	One PT operation per router BOM operation, in sequence order.
<b>Resource Requirements</b>	The machine, pool, and labor slots each operation needs (from Resource Management capability mappings).
<b>Resource Capabilities</b>	The specific capabilities required at each resource slot.
<b>Activities</b>	One internal activity per operation, used by PT for scheduling granularity.
<b>Materials</b>	Material consumption rows derived from the router's BOM.
<b>Products</b>	Output items for each operation, including phantom WIP items between steps.
<b>Operation Attributes</b>	Attribute/code pairs assigned to each operation, used by the PT Optimizer to group or sequence compatible operations (e.g., same burn material class or paint color).
<b>Path / Path Nodes</b>	The linear operation sequence that PT follows when scheduling.



**Resource requirement rows are only generated for operations that have active capability mappings in Resource Management.** An operation with no capability mapping will still appear in the template (with its activity, time, and BOM rows), but PT will not be able to assign a resource to it until a mapping is configured.

“**Operation attribute rows are only generated for operations that have active attribute links configured on the router.** Operations without attribute links will not have attribute rows — PT will not use them for grouping or sequencing until links are added and the template is regenerated.

## 4. The ERP Data Model

All template data lives in the `aps` schema alongside job and resource data, using `source_type = 'TEMPLATE'` to distinguish template rows from live work order rows.

### The Router Link

Table	What it holds
<code>aps.template_router_link</code>	The anchor record tying one router revision to its generated template. Holds the canonical <code>link_id</code> that drives all generated IDs.

The `link_id` is the key that all generated IDs are built from:

Generated ID	Pattern	Example
Template Job ID	<code>template-{@link_id}</code>	<code>template-42</code>
Template MO ID	<code>template-{@link_id}-mo</code>	<code>template-42-mo</code>
Template Operation ID	<code>template-{@link_id}-op-{@bom_op_id}</code>	<code>template-42-op-68664</code>

### Template Source Tables

Template rows share the same source tables as live job data — they are distinguished by `source_type = 'TEMPLATE'`:

Table	Template content
aps.job_source	One template job per router link.
aps.manufacturing_order_source	One template MO per router link.
aps.job_operation_source	One row per BOM operation.
aps.job_resource_source	Resource requirement slots (machine, pool, labor).
aps.job_resource_capability_source	Capability requirements per resource slot.
aps.job_activity_source	One activity per operation.
aps.job_material_source	Material consumption rows.
aps.job_product_source	Output items, including phantom WIP.
aps.job_path_source	The single default path for the template MO.
aps.job_path_node_source	Predecessor/successor pairs defining operation sequence.
aps.job_operation_attribute_source	Attribute/code pairs per operation, generated from active attribute links on the router.

## 5. Using the Templates Screen

The **Templates** tab is located inside the **Product Information** app. It is the central place to view router status, generate templates, and inspect what was generated.

Access to generate templates requires the **Manufacturing Engineering, Product Data Manager**, or **IT Admin** role.

### Main Toolbar

Three buttons appear at the top of the screen:

Button	Behavior
<b>Generate Template</b>	Generates or regenerates the APS template for the selected router. Disabled unless exactly one <b>current</b> router is selected. Prompts for confirmation before running.
<b>Router Details</b>	Opens the Router Details panel for the selected router. Shows the operation list plus any blocking errors or warnings, without committing any data.
<b>Show / Hide Template Details</b>	Expands or collapses the Template Details panel below the Routers grid (see Section 5.4).

---

# Routers Grid

This grid lists all routers and their current template status.

**Columns displayed by default:** Router Name, Part, Rev, Description, Effective From, Effective To, Current, Template Status, Template Job ID, Last Generated.

## Row colors:

Color	Meaning
Yellow	Router is not current (archived). Template generation is blocked.
Red	Router is current but has <b>no template</b> — it has never been generated.
No color	Router is current and has a generated template.

## Routers grid toolbar:

Button	Behavior
Show / Hide Archived	Toggles visibility of non-current (archived) routers. Defaults to hidden — only current routers are shown on load.
Show Only Unmapped / Show All	Filters to routers with no template at all ( <code>Template Status = none</code> ). Use this to find gaps before a PT go-live.

## Template Status values:

Status	Meaning
<code>none</code>	No template has ever been generated for this router.
<code>generated</code>	A template exists and was generated from this router. Check <b>Last Generated</b> to see when.

---

# Router Details Panel

Clicking **Router Details** opens a modal for the selected router without modifying any data. It shows:

- **Blocking Errors** (red) — Problems that will prevent generation from succeeding. Fix these before attempting to generate. The most common is that the router is not current.

- **Warnings** (yellow) — Non-blocking issues that generation will proceed despite. The most common is that one or more operations have no active capability mapping in Resource Management, meaning those operations will not receive resource requirement rows.
- **Router Operations table** — Lists every BOM operation in sequence with setup and run times, its generated Template Operation ID (if a template already exists), and two readiness flags:
  - **Resource Mapped** — Whether an active capability mapping exists for this operation in Resource Management. Operations without a mapping will not have resource slots generated.
  - **Attribute Mapped** — Whether the operation has at least one active attribute link configured in the Routers tab. Operations without attribute links will not have attribute rows generated and will not be used by the PT Optimizer for grouping or sequencing.

“ Use the Router Details panel to check readiness before generating, especially for routers whose operations have recently changed or whose capability mappings are new.

## Generation Result Panel

After clicking **Generate Template** and confirming, a result panel appears showing:

Field	What it means
<b>Template Job ID</b>	The PT job ID that was created or reused for this template.
<b>Template MO ID</b>	The PT manufacturing order ID for this template.
<b>Created</b>	Number of new rows inserted across all template tables.
<b>Updated</b>	Number of existing rows refreshed.
<b>Skipped</b>	Rows that were already up to date and required no change.
<b>Errors</b>	Count of rows that failed. Any errors trigger a rollback — nothing is partially committed.
<b>Warnings</b>	Non-blocking notices, such as operations skipped due to missing capability mappings.

A green success banner also appears at the top of the screen summarizing the result. The Routers grid refreshes automatically after generation to reflect the updated **Last Generated** date and **Template Status**.

# Template Details Panel

Clicking **Show Template Details** expands a set of read-only inspection tables below the Routers grid. These tables show the raw APS source rows that were generated for a template — useful for verifying what PT will receive.

Selecting a router row (when the panel is visible) automatically loads that router's template data across all tables:

- Template Job, Manufacturing Order, Operations, Paths, Path Nodes
- Activities, Resources, Resource Capabilities
- Materials, Products, Operation Attributes

A **Load All Templates** button on the Template Job table loads every template job in the system for comparison.

“ This panel is primarily a diagnostic tool for verifying generation output and troubleshooting scheduling issues. Normal template maintenance does not require using it.

## 6. When to Generate or Regenerate

Situation	Action
A new current router has no template (red row).	Select it and click <b>Generate Template</b> .
A router revision was superseded by a new revision.	Select the new current router row and generate. The old revision's template remains but will no longer be used for new jobs.
Capability mappings were added or changed for an operation.	Regenerate the template for any affected router so the new resource requirement rows are picked up.
Generation produced warnings about unmapped operations.	Configure the missing capability mappings in Resource Management, then regenerate.
Attribute links were added or changed for an operation.	Regenerate the template for any affected router so the updated attribute rows are picked up by PT.
Generation produced warnings about operations with no attribute mapping.	Add attribute links for those operations in the Routers tab, then regenerate.

Situation	Action
Troubleshooting a scheduling issue in PT for a specific part.	Use <b>Router Details</b> to check readiness flags, then use <b>Show Template Details</b> to inspect the generated rows.

## 7. Phase 1 Scope: What's Used and What's Not

Feature	Phase 1 Status	Notes
Template generation from current routers	<input type="checkbox"/> Used	One template per current router. Non-current routers are blocked.
Resource requirement generation	<input type="checkbox"/> Used	Driven by capability mappings in Resource Management. Operations without mappings are skipped.
Material and product generation	<input type="checkbox"/> Used	Derived from the router's BOM. Phantom WIP items are generated automatically.
Path and path node generation	<input type="checkbox"/> Used	Single linear path per template MO.
Operation attribute generation	<input type="checkbox"/> Used	Generated from active attribute links on the router's BOM operations. Operations without attribute links produce no attribute rows and will not be used by the PT Optimizer for grouping or sequencing.
Sub-assembly MO expansion	<input type="checkbox"/> Not used	One MO per template; no recursive sub-assembly expansion.
Automatic regeneration on router publish	<input type="checkbox"/> Not used	Manual regeneration only in Phase 1. Automation is a future enhancement.
Template deletion / deactivation	<input type="checkbox"/> Not used	Old templates from archived routers are not automatically retired.

# Storage Areas

**Audience:** ERP users who are familiar with Action Target's system but are new to PlanetTogether.

**Scope:** Phase 1 — storage area management, cell groups, and the item storage coverage

requirement. **Related stories:** US-005 (Inventory and Warehouse Maintenance Matrix), US-006 (Inventory and Warehouse Maintenance Tools).

---

## 1. How PlanetTogether Uses Storage Areas

PlanetTogether requires three objects before it will accept an item during import:

1. An **Items** row — the item master record.
2. An **Inventories** row — the item's inventory definition for a warehouse.
3. At least one **ItemStorage** row — a mapping from that item/warehouse pair to a specific storage area.

If any of the three is missing for an exported item, PT will reject the item *and* any sales orders, work orders, shop jobs, or job materials that reference it. Storage areas are therefore not optional configuration — they are a gating requirement for every item that will be scheduled in PT.

The ERP manages storage area data in the `aps` schema. PT reads the data through export views; it does not write back to storage area records in Phase 1.

---

## 2. The Three Storage Area Types

Every storage area has an **ERP Type** that describes how it is linked to the ERP:

Type	What it represents	Storage Area ID generated as
<b>Location</b>	Linked to a specific ERP material location ( <code>mat_loc_cell</code> ). Use when one location's inventory maps cleanly to one PT storage area.	<code>LOC-<code>{location_id}</code></code>

Type	What it represents	Storage Area ID generated as
<b>Cell Group</b>	Linked to a named set of bank/column/row ranges ( <code>aps.cell_groups</code> ). Use when physical cells defined by a cell group represent a single logical storage area.	<code>CELLS-{cell_group_slug}</code>
<b>Placeholder</b>	No ERP link. Used for default/unmapped areas such as <code>UNMAPPED</code> or <code>AMS_PLACEHOLDER</code> .	The storage area name itself

“ **Storage area IDs are generated automatically by a database trigger.** You supply a name when creating the area; the system assigns the ID based on the type and linked object. Once created, the ID is frozen — it cannot change even if the name is edited later.

“ **A storage area cannot link to both a location and a cell group at the same time.** The system enforces an either/or (XOR) rule. Selecting a different ERP Type in the create form automatically clears the other field.

## The Default / Unmapped Placeholder

Every active warehouse should have at least one **placeholder** storage area that acts as the unmapped default. When an item/warehouse pair does not yet have a business-owned storage area assignment, the export system can assign it to this default row so PT can still import the item. Without a default, those items fail the ItemStorage requirement and block the export.

Use the **Validate / Readiness** check (see Section 5) to confirm each warehouse has a default before exporting.

## 3. Cell Groups and Mappings

A **Cell Group** is a named, reusable set of physical cell ranges. It exists independently of any single storage area and can be referenced by exactly one active storage area at a time.

Each cell group has:

- A **Name** — a human-readable label (e.g., "Weld Cells").

- A **Slug** — a short uppercase identifier used in the auto-generated storage area ID. Must be uppercase letters, digits, and hyphens only — no spaces. Examples: `WELD-CELLS`, `ASSEMBLY-A`.

**Cell Group Mappings** define which physical cells the group covers. Each mapping row specifies:

- **Bank** — the racking bank identifier (e.g., `101A`).
- **Col From / Col To** — the column range within that bank.
- **Row From / Row To** — the optional row range within those columns.

One cell group can have many mapping rows — one per bank or contiguous range. The union of all mapping rows describes the full set of cells the group covers.

## 4. The ERP Data Model

All storage area data that flows to PlanetTogether is owned in the `aps` schema. PT reads from **export views** (`aps.*_v`) that filter and shape the source tables.

### Source Tables

Table	What it holds
<code>aps.storage_area_source</code>	Storage area master records. Holds <code>erp_type</code> , <code>erp_inv</code> , <code>mrp_nettable</code> , <code>constrained</code> , <code>erp_location_id</code> , and <code>erp_cell_group_id</code> . The <code>storage_area_id</code> column is trigger-assigned on insert and cannot be changed afterward.
<code>aps.cell_groups</code>	Cell group master records. The <code>slug</code> column drives the <code>CELLS-<code>{slug}</code></code> storage area ID.
<code>aps.cell_group_mappings</code>	Bank/column/row range rows for each cell group.

### Export Views (what PT reads)

View	Feeds PT object	Notes
<code>aps.storage_areas_v</code>	<code>Inventories</code> / <code>ItemStorage</code> reference	Active storage areas only.
<code>aps.item_storage_v</code>	<code>ItemStorage</code>	Active item/storage area pairs only.
<code>aps.item_storage_lots_v</code>	<code>ItemStorageLots</code>	Active lot-level assignments.

The ERP is the authoritative source. Changes made inside PT to storage area data are not written back in Phase 1.

---

# 5. Using the Inventory Manager Setup Screen

The **Setup** tab in the **Inventory Manager** app is the central place to maintain storage areas and cell groups.

Access to create, edit, and retire records requires the **Quality Manager** role.

The screen operates in two modes controlled by toolbar toggles. The default mode when you open the tab is **Storage Areas**.

---

## Main Toolbar

Button	What it does
Show / Hide Storage Areas	Toggles the Storage Areas list. Visible by default.
Show / Hide Setup Tools	Toggles the Setup Tools section (cell groups and mappings). Hidden by default.
Validate / Readiness	Runs a live readiness check and opens the results modal.
Preview Ensure Item Storage	Shows which item/warehouse pairs would receive a default storage area assignment if the ensure function were run.
Refresh	Reloads the storage areas list from the database.

---

## Storage Areas List

This grid shows all storage areas from `aps.storage_area_source`.

**Columns displayed:** Storage Area ID, Name, Warehouse, ERP Type, ERP Inv, Location / Cell Group link target, Default, Default-only Item Count.

**Toolbar actions:**

Action	Behavior
Create Storage Area	Opens the create modal.

Action	Behavior
<b>Edit</b>	Opens the edit modal for the selected row. Requires exactly one row selected.
<b>Retire</b>	Prompts for confirmation, then marks the storage area inactive. Records are never physically deleted. A default placeholder cannot be retired if it is the only default for its warehouse.

## Create / Edit Storage Area Modal

Both create and edit use the same form. In edit mode, the Storage Area ID is shown as read-only — it was assigned at creation and cannot change.

### Form fields:

Field	Required	Notes
Storage Area Name	<input type="checkbox"/>	The display label in PT and in the ERP.
Warehouse	<input type="checkbox"/>	Autocomplete from active warehouses.
ERP Type	<input type="checkbox"/>	<input type="text" value="Location"/> , <input type="text" value="Cell Group"/> , or <input type="text" value="Placeholder"/> . Changing type resets the other link field.
Inventory Class	<input type="checkbox"/>	<input type="text" value="INV"/> , <input type="text" value="WIP"/> , or <input type="text" value="AMS"/> . Defaults to <input type="text" value="INV"/> .
ERP Location	Context	Shown when ERP Type is <input type="text" value="Location"/> . Autocomplete from active ERP material locations.
Cell Group	Context	Shown when ERP Type is <input type="text" value="Cell Group"/> . Autocomplete from active cell groups, sorted by slug.
MRP Nettable		Checkbox. Defaults to checked. Controls whether PT includes this area's stock in net requirements.
Constrained		Checkbox. Defaults to unchecked.
Notes		Optional free text. Not sent to PT.

“ **ERP Location and Cell Group are mutually exclusive.** Only one field is visible at a time based on ERP Type. Switching types clears the previously chosen value.

# Validate / Readiness Modal

Click **Validate / Readiness** in the main toolbar to open this modal. It displays four counters plus a row-level issue list.

Counter	What it flags
Warehouses Missing Default	Warehouses with no active placeholder storage area marked as default.
Items Missing Storage	Item/warehouse pairs with no <code>ItemStorage</code> row — these will fail PT import.
Inactive Location Links	Storage areas of type <code>Location</code> whose linked <code>mat_loc_cell</code> row is no longer active.
Inactive Cell Group Links	Storage areas of type <code>Cell Group</code> whose linked cell group is no longer active.

The issue list provides row-level detail: the issue code, a plain-language message, and the affected Storage Area ID if applicable.

This modal is read-only. Resolve issues by correcting the underlying storage area, location, or cell group data, then run the check again.

---

## Preview Ensure Item Storage Modal

Click **Preview Ensure Item Storage** to see which item/warehouse pairs currently lack an `ItemStorage` row. The modal shows the row count that would be inserted and a table of the affected item/warehouse/storage area combinations.

“ **This is preview-only.** No data is written when you open this modal. The actual ensure operation requires a separate approval step and is run through a database function before export.

Use this preview as a pre-export check to confirm all items are covered before kicking off a PT import.

---

## Setup Tools — Cell Groups

Click **Show Setup Tools** in the main toolbar, then **Show Cell Groups** to manage cell groups and their mappings.

### Cell Groups grid toolbar:

Action	Behavior
<b>Create Cell Group</b>	Opens the cell group form in create mode.
<b>Edit</b>	Opens the form for the selected cell group.
<b>Retire</b>	Marks the cell group inactive. Blocked if an active storage area currently references this cell group.
<b>Show All (incl. Inactive) / Show Active Only</b>	Toggles visibility of inactive cell groups.

### Cell group form fields:

Field	Required	Notes
Name	<input type="checkbox"/>	Human-readable label.
Slug	<input type="checkbox"/>	Uppercase letters, digits, and hyphens only. Drives the <code>CELLS-{slug}</code> storage area ID. Must be unique.
Description		Optional.
Notes		Optional internal notes.

### Cell Group Mappings grid:

Selecting a cell group row filters the Mappings grid to that group's rows. Each mapping covers one bank/range.

Action	Behavior
<b>Add Mapping</b>	Opens the mapping form. Select a bank and enter column/row ranges.
<b>Retire</b>	Removes the mapping row from the active set.

Mapping form fields: **Bank** (autocomplete from ERP racking banks), **Col From**, **Col To**, **Row From** (optional), **Row To** (optional), **Notes** (optional).

## 6. Phase 1 Scope: What's Used and What's Not

Feature	Phase 1 Status	Notes
---------	----------------	-------

Storage area create / edit / retire	<input type="checkbox"/> Used	Full CRUD via Inventory Manager Setup tab.
All three ERP Types (Location, Cell Group, Placeholder)	<input type="checkbox"/> Used	All supported.
Cell group create / edit / retire	<input type="checkbox"/> Used	Managed in Setup Tools.
Cell group bank/column/row mappings	<input type="checkbox"/> Used	Add / retire mapping rows per cell group.
Validate / Readiness check	<input type="checkbox"/> Used	Four-counter check plus row-level issue list.
Preview Ensure Item Storage	<input type="checkbox"/> Used	Pre-export coverage check.
ERP location editor (create/edit ERP locations)	<input type="checkbox"/> Deferred	Locations are read-only lookup targets in Phase 1.
Racking cell / cell range preview	<input type="checkbox"/> Deferred	Phase 2 Setup Tools expansion.
Legacy Material setup functions	<input type="checkbox"/> Deferred	Ported from Tcl in a later phase.
PT write-back (PT → ERP storage area changes)	<input type="checkbox"/> Not in scope	ERP is authoritative; PT is a read-only consumer for Phase 1.

## 7. Summary: The Minimum You Need Before Items Can Export

Before PlanetTogether can successfully import item and inventory data, the following must be true:

- Every active warehouse has at least one **default placeholder** storage area.
- Every active storage area of type `Location` links to a currently active ERP material location.
- Every active storage area of type `Cell Group` links to a currently active cell group with at least one mapping row.
- Every item/warehouse pair that will be exported has at least one `ItemStorage` row — confirm with **Preview Ensure Item Storage** before running the export.
- Validate / Readiness** shows zero issues in all four counters.

Missing any of these will cause PT to reject the affected items and all dependent sales orders, work orders, and job materials during import.