Case Study | Manufacturing | OpsHub x SPK and Associates

Unifying ALM and engineering

Codebeamer–Azure DevOps integration for ALM continuity

Global energy storage solutions leader

A global leader in stored energy solutions, needed to unify its engineering workflows across Codebeamer and Azure DevOps. OpsHub Integration Manager (OIM) partnered with SPK and Associates to deliver a bi-directional, rule-driven digital thread, enabling controlled sync of Epics, Features, User Stories, and Test entities across both platforms with zero workflow disruption

2 Core platforms integrated

0 Manual data recollection

1 Unified digital thread established

01 The challenge

Siloed tools · Sync complexity · Workflow friction

01 Siloed ALM & DevOps ecosystems

The Company operated Codebeamer as its requirements and test management hub while Azure DevOps drove engineering execution. The two platforms functioned as independent silos, preventing a unified view of development progress and traceability

02 Complex entity hierarchy to preserve

Company's work hierarchy; Epics, Features, Stories, Test Cases, Test Sets, and Test Runs needed to be translated accurately between Codebeamer and ADO entity models, preserving parent-child relationships and "Tests/Verifies" traceability links

03 No controlled sync trigger mechanism

Teams needed selective synchronisation; not every Codebeamer Epic should flow into ADO. The absence of a configurable sync-trigger field (such as "ADO Sync") meant all-or-nothing integration, which was unacceptable for company's phased delivery model

04 Risk of unwanted data modification

Fields like Start Date, Target Date, and Efforts were to be managed in ADO and synced back. Without proper field-level governance, engineers risked overwriting authoritative ADO data from Codebeamer, or vice versa, causing inconsistency.

05 Test entity sync with hierarchy

Test Suites in ADO needed to map to Test Sets in Codebeamer while preserving the hierarchy. Test Runs and Test Results also had to synchronise as Test Runs in Codebeamer, adding another layer of multi-directional complexity.

06 Link update detection limitations

Neither Codebeamer nor ADO updates an entity's "last modified" timestamp when only links or references change. This made it impossible to detect and propagate link updates (e.g., "Tests" relationships) without additional custom automation.

02 The OIM solution

Intelligent, hierarchy-respecting data sync

Execution ready ADO pipeline

A custom ADO Sync flag in Codebeamer ensures Epics only flow into Azure DevOps when they hit "Ready to Develop" state, keeping the execution pipeline clean and intentional.

No disruption to ongoing work

Bi-directional sync for Epics and Stories, uni directional for Features and Test entities, this was structured exactly around how company's teams author and own work, with no duplication or rework.

Hierarchy & parent-links intact

Feature's parent in ADO translated to a "Satisfies" link in Codebeamer. The same logic applied to User Stories, ensuring architectural traceability is never lost during sync.

Full test visibility in Codebeamer

Test Cases created in ADO were synced to Codebeamer as Software Tests. Test Suites mapped to Test Sets (with hierarchy preserved). Delivering end-to-end test traceability across both platforms.

Clear data ownership across systems

OIM ensured that Start Date, Target Date, and Efforts are governed exclusively in ADO and synced back to Codebeamer. Without any conflicting edits, or overwriting.

All link changes preserved

When links are added or removed in ADO, a custom rule would trigger OIM to pick up the change - so relationship updates always propagate across both systems.

03 Integration architecture

Digital thread · Bi-directional sync · On-premises + cloud

codebeamer — ALM · Requirements ⟷ Azure DevOps — Work tracking and engineering execution

OpsHub integration layer

OpsHub Integration Manager (OIM)

Digital thread | Rule-driven sync | Bi-directional integration

Controlled delivery

Only Epics that are ready for engineering move from Codebeamer to Azure DevOps. ADO Sync flag ensures that incomplete requirements never enter the dev backlog.

Full traceability

Every work item remains connected across both tools. Parent relationships and links are translated automatically so teams always see complete lifecycle traceability

Trusted data

Teams work with consistent and reliable planning data across both systems, ensuring effort, and execution information remains accurate without conflicting updates.

04 Integration execution

Tailored strategy · Phased · Zero disruption

INTEGRATION DELIVERY PHASES

Discovery — Ecosystem audit
- Full infrastructure audit
- Mapping of instances
- Custom field requirements defined
- Entity hierarchy documented

Strategy — Tailored design
- Sync direction per entity type defined
- ADO Sync trigger logic architected
- Entity model mapping finalised
- Bi-directional sync mapping

Implementation — OIM deployment
- OIM deployed on SPKAA-managed Ubuntu + Windows infrastructure
- CB -ADO connectors configured and validated
- Custom ADO rules deployed
- Test & Production instances configured in parallel

Validation — Compliance check
- Sync verified across all entities
- Hierarchy preservation confirmed
- Test entity sync and link propagation validated
- Stakeholder sign-off obtained

05 Integration scope

Full entity coverage across both tools

Codebeamer

ALM · Regulatory reporting

Role in Integration	Central Hub
Sync Type	Bi-directional (Epics, User Stories)
Change Traceability	End-to-End
Deployment Type	On-Premises + Cloud
Connected Entity Types	Requirements, Test cases, Epics, Features, Test runs

Integration Summary

Bi-directional · Real-time · Hybrid

Total Tools Integrated	2
OIM Hosting	SPKAA-managed Ubuntu + Windows
Database	PostgreSQL 16.2

06 Engagement outcomes

Results at a glance

2 Tools connected via digital thread

0 Manual compliance consolidation effort

100% Change traceability across all systems

0 Disruption to ongoing workflows

Before integration
- Both tools operating as isolated silos
- Manual effort to keep work item hierarchies aligned
- Test entity lifecycle managed separately
- Risk of unintended field overwrites between tools

OpsHub x SPK and Associates

After integration
- Unified digital thread across Codebeamer and Azure Devops
- Only approved Epics move into Azure DevOps for development
- Epic → Feature → Story hierarchy preserved across systems
- Link and traceability updates sync automatically

07 OIM capabilities leveraged

What made this integration possible

Capability	Description	Integration Role
Bi-directional sync	Both uni-directional and bi-directional integration between Codebeamer and Azure DevOps	Ensures any change in any tool is reflected across both tools in real-time
Hierarchy preservation	Parent-child relationships translated across both systems	Maintains Epic → Feature → Story traceability
Test entity lifecycle sync	Test Cases, Test Sets, and Test Runs synced from ADO to Codebeamer	Provides automated and full visibility of test execution in Codebeamer
Source-of-truth governance	Each system owns specific data domains such as scheduling or requirements	Prevents conflicting updates and maintains trusted data
Continuous synchronization	Changes propagate automatically once items enter the digital thread	Keeps both platforms consistently aligned
Reliable relationship updates	Cross-entity links and traceability relationships stay synchronized	Empowered leadership with real-time traceable insights into development activities and cross-functional progress

08 Engagement benefits

What was delivered

01 Controlled, intelligent synchronisation

The ADO Sync flag ensures only approved Epics move from Codebeamer into Azure DevOps. Teams can refine requirements in Codebeamer before development begins, preventing incomplete work from entering the dev backlog.

02 End-to-end traceability

Every work item, from Epic to Test Result, remains connected across Codebeamer and Azure DevOps. Parent relationships and traceability links such as Satisfies and Verifies are preserved automatically across both platforms.

03 Preserved field-level data integrity

Scheduling fields such as Start Date, Target Date, and Effort are governed in Azure DevOps and synced back to Codebeamer. Field restrictions ensure each system remains the authoritative source for its data.

04 Unified test lifecycle visibility

Test Cases created in Azure DevOps, organized into Test Suites, and executed through Test Runs are fully visible in Codebeamer. This provides quality and compliance teams a complete view of test coverage and results.

05 Reliable link & traceability propagation

A custom Modified Date automation ensures link updates in Azure DevOps trigger synchronization. This allows test relationships and traceability links to propagate reliably across both systems.

06 Zero workflow disruption at scale

Teams continue working in their preferred tools. Codebeamer manages requirements and compliance, while Azure DevOps supports development execution. OIM synchronizes both environments without changing existing workflows.

09 Integration roadmap

Delivered, live & next steps

Phase 1: Foundation
- → Full environment audit of CB v2.2.0.1 and ADO cloud instances
- → Tailored digital thread integration strategy designed
- → OIM deployed and configured for both tools

Phase 2: Live integration
- → Bi-directional sync active across both tools
- → ADO Sync flag trigger validated end-to-end
- → Link propagation confirmed in production
- → Hierarchy preserved across both systems
- → Cross-functional silos fully eliminated

Phase 3: Optimization
- → Authentication upgrade from Basic to SAML 2.0 for production OIM
- → Expanded field-level sync rules and additional custom fields
- → Additional entity types and deeper reporting integration

"

By leveraging the joint capabilities of OpsHub and SPK and Associates, company transformed two independent ALM platforms into a single, coherent digital thread. Teams in Codebeamer and Azure DevOps now operate with full visibility across the entire delivery lifecycle without changing their native workflows. The partnership delivered exactly the integration precision that company's global engineering operations demanded.

OPSHUB

Let's Talk