



Accelerating S/4HANA Consolidation with Test Automation In Only 7 Weeks

Petter-Andre Johansson and Michael Wolff | Pearl Group | SAPSA 2026

Pearl Group × Bohus

What We Will Cover

- 01 The Challenge**
Why Bohus needed test automation for consolidation
- 02 Why Automate?**
The cost curve, the 80/20 rule, and regression risk
- 03 UiPath for SAP**
Platform overview, Test Manager, and architecture
- 04 How We Built It**
Modular design, smart arguments, data-driven testing
- 05 Lessons & Takeaways**
Honest challenges, scaling effect, and what we learned

Bohus Consolidation — The Context

63+

stores

Separate entities
consolidated into one

7

weeks

Total timeline for getting UiPath test
automation ready

EWM

Extended Warehouse
Management included on 5th January

Bohus needed to consolidate 63+ individual store entities into a unified S/4HANA structure — including EWM — under extreme time pressure. We started 17th of September and were technically ready before Christmas (19.12.26).

For the rollout in February, manual regression testing was not an option. The volume of processes to validate after each rollout wave made automation essential.

Why Automated Testing Matters

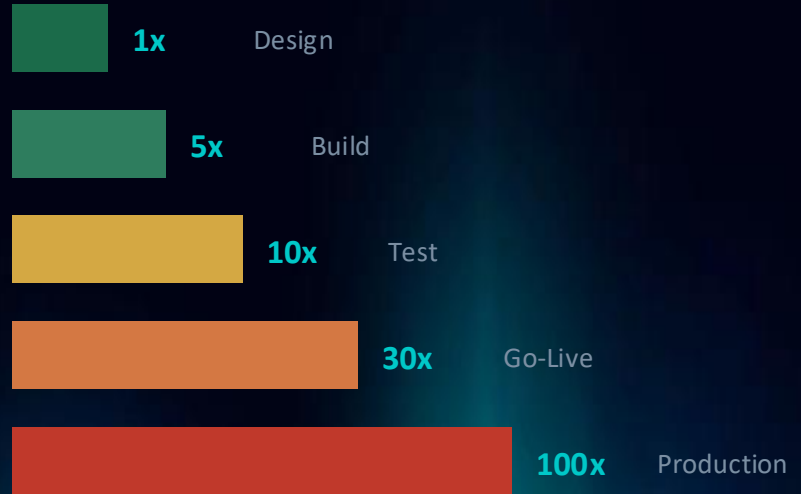
Every SAP project carries regression risk: transports, config changes, upgrades

Manual regression testing does not scale under go-live pressure

Automated testing = faster feedback, consistent coverage, audit-ready evidence

Defects found late are significantly more expensive to fix

Cost of Fixing Defects by Phase



The question is not whether to automate — it is which tool for which scenario

The 80/20 of SAP Testing

"20% of SAP processes drive 80% of business risk"

Today: Manual Testing

- Testers spend 80% of time on routine, repeatable regression scenarios
- The critical 20% — complex edge cases — gets less attention
- Errors in the 20% cause 80% of production incidents
- Resources trapped in repetition, not applied to complexity

With UiPath: Automated Testing

- The repetitive 80% runs automatically — most common scenarios covered
- Testers focus entirely on the complex 20% requiring human judgment
- Edge cases and exploratory testing get attention they deserve
- Production incidents drop — the routine is never skipped

What is UiPath?

Enterprise automation platform trusted by 10,000+ organisations. Automates repetitive, rule-based tasks across any system.

Automation

- SAP GUI, Fiori, Web, API, Desktop
- Scheduled or on-demand execution
- Full audit trail and evidence capture

Testing

- Regression testing with evidence
- Full traceability: requirement → test → result
- CI/CD integration for SAP transports

Agentic AI

- Generates test cases from requirements
- Self-heals broken tests at runtime
- Analyses results and surfaces root causes

UiPath Test Manager

Centralised platform for test planning, execution, and reporting — with AI agents at every stage

Plan & Manage

Link requirements to test cases. Organise test sets. Track coverage and traceability.

Automate

Build automated test cases in Studio with low-code or coded workflows. SAP GUI, Fiori, Web, API.

AI Agents

Autopilot generates test cases. Healing Agent repairs broken selectors. Root cause analysis.

Report & Analyse

Real-time dashboards, live test streaming, evidence with screenshots. Stakeholder-ready reports.

Where Does Automated Testing Fit?

Test automation built during Realize is an asset for the entire system lifecycle

Explore	Realize	UAT	Go-Live	BAU
<ul style="list-style-type: none"> Define test scope and strategy 	<ul style="list-style-type: none"> Build tests alongside development 	<ul style="list-style-type: none"> Run automated regression 	<ul style="list-style-type: none"> Full regression before cutover 	<ul style="list-style-type: none"> Regression after every transport
<ul style="list-style-type: none"> Identify standard vs custom processes 	<ul style="list-style-type: none"> Establish regression baseline 	<ul style="list-style-type: none"> Convert manual scripts to automation 	<ul style="list-style-type: none"> Automated smoke tests post go-live 	<ul style="list-style-type: none"> Quarterly upgrade validation
<ul style="list-style-type: none"> Decide tooling: TAT, UiPath, or both 	<ul style="list-style-type: none"> Automate custom transactions 	<ul style="list-style-type: none"> Traceability captured automatically 	<ul style="list-style-type: none"> Fast feedback on critical processes 	<ul style="list-style-type: none"> Automation library grows with system

The earlier automation is introduced, the greater the return across the full project lifecycle

How We Built It

Pearl Group's approach to building test automation that scales

The Three-Layer Module Architecture

TEST CASE LAYER *TC_O2C_SalesOrder_Create.xaml*

Orchestrates. Reads data. Calls business module. Asserts result.

Responsible for: What to test and with what data

BUSINESS LAYER *O2C_SalesOrder_Create.xaml*

Executes one business action — Create Sales Order, Post Goods Issue, Create Invoice.

Responsible for: How to perform the business action

TECHNICAL LAYER *SAP_NavigateToTcode.xaml*

Low-level SAP building blocks — login, navigate, error handling, read status bar.

Responsible for: How to talk to SAP

When SAP changes: fix the Technical layer. When the process changes: fix the Business layer. The Test Case layer almost never changes.

Arguments — The Smart Way to Handle Variations

Every test case module has three types of inputs:

Mandatory

Must be provided — genuinely different for every test

CustomerNumber, Material, Quantity

These are what you're actually testing — they must vary

Optional with Default

Has a sensible default — only override when testing something specific

OrderType = "OR", Currency = "NOK"

80% of tests use the default. 20% override for specific scenarios.

Config-Sourced

Never passed as an argument — comes from environment configuration

SalesOrg, CompanyCode, SAP Connection

Same for all tests in the same environment. Never hardcoded.

Data-Driven Testing — One Module, Many Scenarios

1. Define test data

2. Load at runtime

3. Execute module

4. Capture result

Scenario	Customer	Material	Qty	Expected	Result
Standard SO	10001	MAT-001	10	created	Pass
Large quantity	10001	MAT-001	9999	created	Pass
Blocked customer	99999	MAT-001	10	blocked	Pass
Invalid material	10001	INVALID	10	does not exist	Pass

Four different scenarios. Zero new code. The module handles all of them — because it was designed to accept data, not assume it.

Challenges — What You Should Know

Learning Curve

UiPath requires a trained developer. Initial slow progress is expected — the first modules take significantly longer than subsequent ones.

No Pre-Delivered SAP Scripts

Unlike SAP Cloud ALM TAT, everything must be built. This is an investment that pays off in flexibility, but the initial effort is real.

Licensing Cost

Studio, Robot, Orchestrator, and Test Manager carry additional cost. A single license investment can span multiple customer engagements — as consultants with access to multiple SAP environments, the same framework and tooling serves across projects.

Customer Adoption

Moving from manual UAT to automated testing changes the customer's role. Knowledge transfer and change management are critical.

Internal Change

Shifting from manual testing services to automated delivery requires new competencies and willingness to change how services are priced.

Lessons Learned — What We'd Do Differently

1

Define Scope First

Create test cases in Test Manager before building anything in Studio. Requirements first, automation second.

2

Get Structure Right

Module structure must be correct from day one — a poor structure makes reuse and maintenance difficult throughout.

3

Know the Process

Process knowledge is critical when building scenarios. Gaps lead to suboptimal modules and costly rewrites.

Effort & Scaling Effect

~16h

First scenario

Includes establishing the module foundation

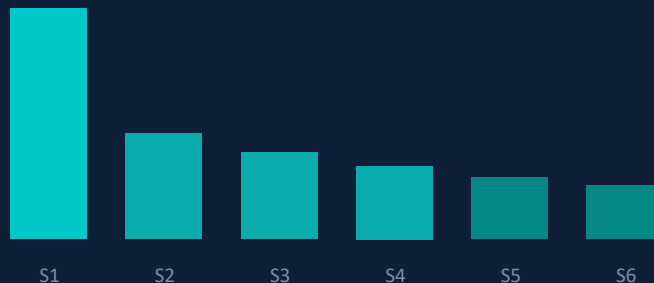
~4h

Each subsequent scenario

Modules are reused across scenarios

The Scaling Effect

The more scenarios we build, the lower the cost per scenario becomes. Modules built once are reused across every new scenario — effort drops significantly over time.



What now from Pearl?

WHY IT MATTERS Manual testing does not scale.

PearlCare's **QA as a Service** uses UiPath automation to reduce testing time, increase quality, and accelerate go-live.

1 The Value Delivered

- › Reduced testing time — automated scripts run in minutes
- › Increased quality — consistent, repeatable execution
- › Production stability — defects caught before go-live
- › Faster go-live — fewer delays, more confidence
- › Pre-built templates — common processes ready from day one

2 How It Works

- › Pre-built templates reduce build time significantly
- › Well-described test cases = faster implementation
- › Automated regression coverage per release
- › Evidence captured automatically — audit-ready UAT sign-off
- › Defects found in test, not production (30–100x cost saving)

3 Two Entry Points

Entry A — Hand-over from project

- › Project built UiPath automations
- › PearlCare takes ownership post go-live

Entry B — Request to build

- › PearlCare creates automations from test cases
- › Better test steps = faster build, lower cost — faster innovation

Key Takeaways

- 01** Test automation is not optional in S/4HANA projects — the question is when, not if
- 02** UiPath and SAP Cloud ALM TAT are complementary — use both where they fit
- 03** Modular architecture (three layers) makes automation reusable and maintainable
- 04** Start automation during Realize — it pays off through UAT, go-live, and BAU
- 05** Invest in process knowledge and structure before writing the first test
- 06** The scaling effect is real — first scenario is expensive, every next one is fast

ABOUT PEARLCARE

PearlCare is Pearl's Application Management Service (AMS) that guarantees the availability and continuous upgrading of your systems.

- **Runs automated UAT testing** using UiPath scripts against every release
- **Maintains pre-built test templates** covering common SAP business processes
- **Captures evidence & traceability** automatically for audit-ready sign-off
- **Takes ownership post go-live** — running and managing test automations ongoing
- **Builds test automations from scratch** based on customer test cases

Want to know more? Contact us today!

Per Bengtsson

MD PearlCare

+46 736336257

per.bengtson@pearlgroup.se

www.pearlgroup.no

Susanna Andersson

Head of Customer Operations

+46 739849260

susanna.andersson@pearlgroup.se

www.pearlgroup.no

Magdalena Hamilton

Portfolio Manager Sweden

+46 729669307

magdalena.hamilton@pearlgroup.se

www.pearlgroup.no

And please share your feedback in the app 😊



Thank You

Questions & Discussion

Petter-André Døplads Johansson

Client Executive

+47 9946 9246

Petter-andre.johansson@pearlgroup.no

www.pearlgroup.no

Mikael Wolff

Head of AI

+4528872114

michael.wolff@pearlgroup.dk

www.pearlgroup.no

SAPSA 2026 — Gothenburg