



## Case Study

# Microsoft Dynamics AX 2009 Test Automation

Anil Kumar  
Test Architect  
Think Future Technologies

### About the client

One of the largest Furniture Retail Company in United States having multiple home stores

### Industry

Retail

### Business need addressed

Ensure that with each release of AX, all functionality were regressed properly

### Solution

Manual Testing coupled with Automation Testing

### Business Needs

Client is one of the largest retail companies in United States with more than 500 stores operating worldwide.

Client has been using a heavily customized Microsoft Dynamics AX 2009( retail ) implementation and fairly recently, some of their stores went down after a release was deployed into their production environment. This caused significant revenue loss and unmeasured amount of customer dissatisfaction. From client's own internal analysis, it turned out to be a regression bug that leaked into production environment.

Client identified that automation testing of AX 2009 besides helping to prevent such regression issues in future, will also provide a solution to the challenging problem of verifying proper working of AX application after deployment in different environments that requires significant time efforts when performed manually. Client is a Microsoft shop and decided to use coded UI testing facility in Visual Studio 2012 for automating dynamic tests against AX 2009 UI

### Challenges

Using Coded UI for Dynamics AX automation turned out to be a unique challenge in itself as TFT team found out during proof of concept stage that several of the AX controls are not supported by Coded UI. To compound that challenge, client's QA team had no access to developers who could enable accessibility features on such controls to make them automatable.

Another major challenge for TFT was to acquire a lot of domain knowledge pertaining to different elements of AX like Account Receivables, Account Payables, Store Setup, POS setup & Management, Inventory Management, and more. To add to the woes, QA team did not had a formal QA environment to conduct testing on. All of the testing was being conducted on staging environment, which was controlled directly by developers who in turn, were deploying application builds to staging environment at their own will. Furthermore, there were no formal test cases in place. All that customer had were Standard Operating Procedures that acted as help guides for their retail store users. Besides, client was not using any test case management tool either.

Even with available staging environment, test data was a big problem. Since staging environment was not controlled well, usually customer's CM team had to rebuild entire environment manually, and it used to take couple of weeks to get up and running.

TFT team conducted a thorough analysis for devising a test strategy and plan to satisfy client's requirements.

In order to work collaboratively with Client's development and testing teams and cover wider scope, TFT designated three separate teams for this assignment –

- **Manual Testing Team** - These folks were responsible for coordinating with Subject Matter Experts and design manual tests that would go into Microsoft Test Manager. All available SOPs were scheduled for refactoring.
- **Framework Development**- Resources on this team were assigned the task of evaluating AX 2009 and POS application for automation ability and were tasked to develop framework extensions in order to support automation team.
- **Test Automation Team**- This team's responsibility was to convert manual test cases into automation scripts using coded UI. This team was expected to ensure that all manual steps are covered by automation scripts.

Test automation implementation primarily involved following 3 components-

- **QA Environment** – A separate QA environment was created with support from IT, QA and CM team. Since there was not enough infrastructure available to create a well distributed deployment environment, all components excluding AOS server were implemented on one terminal server and access was given to every team member of this terminal server.
- **Training for Dynamics AX 2009** – TFT arranged training sessions on AX 2009 for all its team members from a third party specializing in AX training. A local environment was created at TFT's premises to help train all involved teams.
- **Test Management (including tool, test design and execution)** – Microsoft Test Manager was selected as a tool of choice as customer already had TFS 2012 implemented and couple of teams were using it.

However, customer's team had little or no idea of the potential that TFS and Visual Studio 2012 possessed in terms of application lifecycle management and improving team's overall efficiency.

Available SOP (Standard Operating Procedures) were refactored to convert into executable manual test cases with consultation and discussions with SMEs. This not only resulted in more clear specifications for tests, but also made it easy to create test automation scripts.

Even manual execution was now done within MTM, so that auditable test results were made available to top executives.

### Test Automation Framework using Coded UI

- An automation test framework architecture was laid out keeping fundamentals in mind, viz. reliability, performance, robustness and simplicity.
- Entire AX application was logically divided in modules, and each form in each module was represented with one UI map.
- Since AX 2009 was never written with testability in design, we had to resort to customized tools to ensure correct UI map hierarchy was captured.
- User actions on forms were mapped to UI Map methods. E.g. Customers.CreateNewCustomer () method. These methods were in turn used by Test Methods.
- Test data links were established between Microsoft Test Manager and automated scripts. This way, whenever test data was changed to a manual test case in MTM, same was immediately available to automation script also. Effectively, manual and automation scripts were made truly interchangeable and dependable.
- Special handlings were written for controls like Grid, AX Calendar, and POS Calendar etc. This required lot of mathematics to go into calculations.

### Next Steps

- Create SCVMM test environment for AX to support automated build-deploy-test workflow,
- Add multiple POS terminals to support test execution in parallel,
- Implement grid like functionality to support parallel execution of test cases

## Business Value Proposition of AX Automation

There is a perception that since AX 2009 is an out of the box application provided by Microsoft, no automation testing is required. But what we've learnt from this case study is:

- AX Automation simplifies regression testing for heavily customized and complex AX applications.
- A lot of end to end data staging scenarios can be automated thus saving business user's time and allowing them to focus on other critical tasks.
- One of the biggest challenges with any ERP solution, including AX is to verify deployments across multiple environments. This can be easily handled using automation.

### About Think Future Technologies

TFT is leading provider of end-to-end IT services and Technology Management. With many pioneering and innovative firsts to its credit TFT has always led the way in powering the exceptional enterprises with continuum of Technology and its implementation.

We are provider of Best-in-class Enterprise Collaboration (ECP), Enterprise Application Services (EAS), QA & Testing and Support services. TFT has deep industry expertise, business strategy and program governance experience along with tools and methodologies for accelerated deployment of Microsoft solutions. TFT's Intellectual property frameworks illustrates unique invention of process combined with deep domain expertise in multiple industry verticals.

For more information, visit <http://www.tftus.com/>

### For more Information

#### Think Future Technologies

JMD Regent Square

Gurgaon, Haryana, India

[www.tftus.com](http://www.tftus.com)

Email: [info@tftus.com](mailto:info@tftus.com)

Phone: +91 124 2807000