Microsoft Dynamics® AX 2012

Certified for Microsoft Dynamics - Software Solution Test Requirements

This paper describes the requirements to pass the Certified for Microsoft Dynamics - Software Solution Test

Date: September 2011

Contact: msdynamics@veritest.com

Microsoft Dynamics®

Table of Contents

CERTIFIED FOR MICROSOFT DYNAMICS - SOFTWARE SOLUTION TEST REQUIREMENTS

1 Introduction

1.1 This Document

The Certified for Microsoft Dynamics® Software Solution Test (SST) Requirements document for Dynamics AX 2012 describes the technical requirements that an Independent Software Vendor (ISV) solution must meet to successfully pass the Software Solution Test for Dynamics AX 2012.

1.2 SST structure

The SST has been structured into the following two sections

1.2.1 Documentation and self-certification requirements

This section contains the requirements related to documentation and self-certification of the ISV solution. Self-certification for a specific requirement involves confirmation from the ISV that they meet that requirement. This indicates that specific testing has been performed by the ISV internally to verify that their solution meets that requirement and subsequent updates to their solution will be compliant with that requirement. ISVs are responsible for the accuracy of the representations, claims and statements they make during the self-certification process.   Microsoft and its test vendors do not verify the ISV's self-certified solution.

1.2.2 External Certification requirements

The external certification section contains all the requirements that will be verified by the test vendor in their testing facility. This will involve installation of the ISV's solution on a test machine and verification of the requirement by a tester.

1.3 Requirement Structure

Each requirement has been organized into the following sub-sections: Type of requirement, Purpose, Verification Steps and Results.

1.3.1 Type of requirement

A requirement can be classified into three types: General, AOT and Managed Code.

1.3.1.1 General

A general requirement applies to both native Dynamics AX 2012 solutions as well as external (managed code) solutions

1.3.1.2 AOT

An AOT requirement (or native Microsoft Dynamics AX requirement) applies to an ISV solution that resides inside the Application Object Tree (AOT). This can be X++ code, SSRS reports or managed code.

1.3.1.3 Managed Code

A Managed Code requirement applies to an ISV solution that is external to Dynamics AX 2012 and integrates to Microsoft Dynamics AX using one of the prescribed methods. Examples of this are – a web application that connects to Dynamics AX, a rich client (C#) solution that connects to Microsoft Dynamics AX or an external BI or reporting solution that integrates with Microsoft Dynamics AX.

1.3.2   Purpose

Purpose of the requirement outlines the reason for having this requirement as part of the SST i.e. how does the customer / partner benefit from an ISV solution meeting this requirement. Each SST requirement has a customer benefit associated with it and that is documented in this section.

1.3.3   Verification Steps

This section documents the details of the requirement. It provides detailed information on how the testing vendor will validate a particular requirement. ISVs should use this section to understand the requirement and the testing methodology that will be used to verify this requirement. By design this section will be fairly thin for documentation and self-certification requirements since the verification process is also fairly simple for those requirements.

1.3.4   Results

The Results section outlines detailed information on what constitutes a PASS and what constitutes a FAIL for a particular requirement. In some cases detailed scenarios with PASS / FAIL indication have been documented for the benefit of the ISV.

1.3.5   General Documentation Requirements

This sub-section only exists for Documentation and Self-certification requirements. This section provides detailed information on what type of documentation is required from an ISV to pass a certain requirement.

2 Testing Process

Microsoft offers ISV software solution testing through an independent third-party test vendor. ISVs register for the test by visiting the test vendor's Web site http://en-gb.lionbridge.com/product-engineering/product-certification/microsoft/dynamics/default.htm. The vendor site contains a description of the test, as well as an application form with a published test fee schedule.

You can make your solution available to the test vendor for testing by using any of the following methods:

- Option 1: Providing the software with installation instructions for the test vendor to install. This will always be required

- Option 2: Sending a virtual server image of a working configuration of the product to the test vendor

- Option 3: Using an interactive Live Meeting session to provide access to a working configuration of the product

Please note that Option 1 is mandatory even if an ISV chooses Option 2 or Option 3.

After you register your software solution and pay the test fee, the test vendor will contact you with detailed information about the testing process you have selected. For processes involving shipping software or virtual server images to the test vendor, you can choose to send the software on media (CD/DVD), upload to an ftp server, or have the test vendor download from your server. If you choose to use Live Meeting to provide access to your solution, the test vendor will contact you to schedule the session

The following requirements are particularly important:

- Pre-qualification is required. You are responsible for making certain that your solution and organization meet the requirements for submitting and maintaining a Microsoft Dynamics-based solution.

- A build and documentation for your solution must be provided; please see the test guidelines listing each specific requirement. Please confirm your delivery method during registration (download from your server, upload to our ftp or ship via standard mail). Your assistance may be required via email, phone or conference call especially if setup and configuration is complex.

- Please indicate which Requirements do not apply to your solution and ensure this is documented.

- We recommend that you include a document or spreadsheet listing each test requirement number and showing how the requirement is met, for example, in document A on page 4, BP tool, F1 help Role Centers etc.

3   Documentation and self certification requirements

3.1   General documentation on the ISV solution

3.1.1   Type of Requirement: General

3.1.2   Purpose
To ensure that the solution that is being tested is understood by partners and customers, and to aid in the certification process, each ISV is asked to provide a package of documentation with the submission of their application to the test vendor.

3.1.3   General documentation requirements

3.1.3.1   Business functionality and key usage scenarios
Descriptions of your solution's primary functionality and usage scenarios help users understand and adopt your product.
Requirements:
- Identify and describe the purpose of the product.
- Identify the roles that your solution is intended to be used by, and provide a summary of the key and supporting functionality that you provide for each role.

Appendix A contains a template for Key Usage Scenarios. ISVs can create their own formats or use the template provided in Appendix A.

3.1.3.2   Integration points with Microsoft Dynamics AX functionality
Descriptions of how your solution integrates with Microsoft Dynamics AX enable our testers to better understand your solution, and customers to understand how adopting your solution may affect their environment.

Requirement: Provide a complete listing of integration points with the Microsoft Dynamics AX 2012 product.

3.1.3.3 Test scenarios

We require that you provide test scenarios to increase the quality of solutions that run in the Microsoft Dynamics AX environment and give the market the assurance that ISV solutions built on Microsoft Dynamics AX meet technical requirements.

Requirements:
- Provide a functional test scenario for each of the high-level features of your solution.
- Provide a functional test scenario for each integration points key Microsoft Dynamics AX 2012 functionality

3.1.3.4 Application extensibility and customization strategy

A technical explanation of the application extensibility and customization of the solution is required. The documentation is more technical than what targets the solution user. Its purpose is to give VARs a solid understanding of the solution to help them customize and sell the solution to customers.

Requirement: Provide a description of how the solution can be customized.

3.1.3.5 Security development lifecycle (SDL) training

Microsoft security best practices require that developers be aware of secure coding practices, including threat modeling and countermeasures. SDL requires that developers must fix critical bugs that compromise security and perform security code reviews. The purpose of this recommendation is to ensure that ISV software developers receive training on secure software development practices.

Requirement: Provide confirmation that developers have completed the following SDL Core Training Classes

- [Introduction to the Microsoft Security Development Lifecycle (SDL)](#)
- [Introduction to Microsoft Threat Modeling](#)
- [Basics of Secure Design, Development, and Test](#)
- [Privacy in Software Development](#)

3.1.3.6 List of countries or regions where the solution is supported

Requirements:
- List the locales that your solution supports
- Provide all end-user documentation in all supported languages
- Provide installation documentation in all supported languages

3.1.3.7 Microsoft Dynamics AX 2012 licensing requirements for solution

Requirement: Provide a list of all Microsoft Dynamics AX 2012 licenses that your solution requires (i.e. type of Named User CALs and Device CALs required to run your ISV solution).

3.1.3.8 Microsoft Dynamics AX 2012 required footprint for solution

Requirements:

- List the Microsoft Dynamics AX modules or components and the full list of objects that are required for your solution.
- If your solution overwrites any standard modules, objects or components, document them, and provide a justification for each.

3.1.3.9   Installation, setup, configuration and uninstall guidelines

The documentation must provide information that allows partners and customers to successfully install, upgrade, and remove your solution in this environment.

Requirements: Provide an installation and configuration guide in your documentation. This document must identify the targeted installer, who may be a partner, customer or consultant.

3.1.3.10   Registry changes made during installation

Requirement: Document all registry changes that your solution makes during installation/implementation.

3.1.3.11 Restrictions to Microsoft Dynamics AX 2012 functionality

If your solution limits or breaks existing Microsoft Dynamics AX functionality or cannot coexist with Microsoft Dynamics AX functionality, you must provide documentation. For example, if a reporting solution is designed for customers who do not use inventory and will not work for a customer who uses standard inventory functionality, then you must explain this limitation.

Requirement: List each conflict, and describe the Microsoft Dynamics feature or function that will not be available after the solution has been installed.

3.1.3.12 Data upgrade procedures and scripts

The documentation must list the names of the upgrade scripts and the tables that each script affects so that customers and VARs can understand what is required. More in-depth documentation is recommended but optional.

Requirements:

- Prepare upgrade scripts.
- Document each upgrade script, including the tables that each script affects.

3.1.3.13 Backup and restore procedures

Documentation must be provided so that a customer or partner must be able to back up and restore the solution and all associated data.

Requirements:

- Provide a document that describes how your solution data can be backed up and restored.
- Backup procedures must also include any solution data that is stored outside Microsoft Dynamics AX.

3.1.3.14 Static analysis and other test results

If your solution does not compy with FxCop, Best Practices, the UX Guidelines or Security tools, you must provide an explanation for each exception.

Requirement: Describe exceptions for FxCop, Best Practices, UX Guidelines, Security tools and any other test results.

3.1.4 Verification steps

1. ISV submits required documentation as described above, using templates where possible.

2. Test vendor reviews documentation for accuracy and completeness, using the documentation where applicable during the test execution phase of the certification process.

3.1.5 Results

❑ Pass   Documentation is provided and is found to be complete and accurate.
❑ Fail   Documentation is missing or proves incorrect during review or during the test execution phase of the certification process.

3.1.6 Resources

Security Development Lifecycle Training http://www.microsoft.com/security/sdl/discover/training.aspx

Essential Software Security Training for the Microsoft SDL - http://go.microsoft.com/?linkid=9733841

SDL Core Training Classes - http://go.microsoft.com/?linkid=9708478

Microsoft Dynamics AX 2012 Business Value licensing
https://mbs.microsoft.com/partnersource/partneressentials/licensingpolicies/businessvaluelicensing

3.2 Document Compliance with the Microsoft Dynamics AX Foundation Framework Extensions

3.2.1 Type of Requirement: AOT

3.2.2 Purpose

Correct usage of the Microsoft Dynamics AX foundation framework extensions is critical to ensuring the integrity of the application, providing better maintainability and supportablity, and to providing a consistent experience for the end user.

Best Practice checks are used to ensure basic uptake of frameworks. In addition to passing Best Practice checks, we require that ISVs be aware of the following guidelines and sign off that their applications have correctly followed them. By signing off, an ISV is certifying that they understand proper usage of the frameworks and have implemented their solutions using the frameworks properly.

3.2.3 Documentation requirements

3.2.3.1 Reports must be implemented by using SQL Server Reporting Services

Your solution can provide reports that range from raw transactional data to a comprehensive cross-company analysis illustrating the health of your customer's business. Analytical reports are expected to be adaptable, graphical, and interactive to help make informed decisions.
Requirements:
- Provide a list of all ISV Reports that includes both AOT name and navigation path.
- Confirm that there are no X++ reports in the ISV solution

3.2.3.2 Event subscriptions

We are asking that you uptake the use of event subscription when customizing to minimize code conflicts and help facilitate easier code upgrades when future versions are released.
Requirement: Identify locations in your code in which you have used event subscriptions, and provide a code sample.

3.2.3.3 Intelligent Data Management Framework (IDMF) templates

Customers deploying a solution must be able to use the IDMF for archiving and purging data.
Requirements:
- List all IDMF templates that have been modified and created by the ISV

- Provide confirmation that all IDMF templates that have been impacted by changes to tables have been updated to work with the ISV solution.

3.2.3.4   Role Centers
A Role Center is the home page for a user in Microsoft Dynamics AX. Use Role Centers to aggregate trends and alerts from the Microsoft Dynamics AX ecosystem so that the user is able to derive insights at a glance. Each role should have a Role Center associated with it. Role Centers typically have four zones and each zone may contain any number of role center controls.

Requirements:
- Confirm that there is only one worklist control per Role Center in a solution.
- Confirm that all analytic content in the solution can be made visible in a Role Center.

3.2.3.5   Workflow
The workflow infrastructure in Microsoft Dynamics AX provides a mechanism to automate business processes and improve productivity. Workflows are implemented on a per business document basis via metadata. Business users create workflows that use this metadata, and business document records can then be submitted for processing according to these workflows. End users interact with running workflows through dedicated user-interface controls and a personalized work list that are accessible from both the rich client and Enterprise Portal. Workflow tracking data is accumulated at runtime, and can be viewed by end users through the workflow history form, as well as processed into an analytical cube which, when viewed in various workflow reports, provide insight into workflow performance characteristics. The resulting analysis can then drive on-going optimizations to the workflows in use.

Requirement: Confirm that all workflows in the solution have been implemented by using the Microsoft Dynamics AX 2012 workflow infrastructure.

3.2.3.6   Application Integration Framework (AIF)
The Application Integration Framework (AIF) enables companies to integrate Microsoft Dynamics AX and communicate with external business processes and partners through the exchange of XML over various transport media. AIF enables both business-to-business and application-to-application integration scenarios.

Requirements:
- Provide a list of all services in the solution.
- Provide a statement that confirms that all integration to Microsoft Dynamics AX 2012 has been done using AIF exclusively (and specifically stating that neither the .NET Business Connector nor direct SQL queries have been used).

3.2.3.7   Security Framework
Security in Microsoft Dynamics AX works on the basis of role-based access control. Roles are defined as a collection of permissions related to job functions within an organization. The permissions that users might have in the system are defined on the basis of their role membership. This greatly simplifies the administration of security which is done in terms of managing a user's role membership. Microsoft

Dynamics AX 2012 ships with 80 roles out of the box that tie to job functions within an organization. An ISV can modify and manage the permissions and a security administrator can fine tune the roles based on the permissions available.

Requirement: Document the approach to extending the security artifacts in Microsoft Dynamics AX 2012 and provide an example that demonstrates this approach.

3.2.3.8   ISV Licensing Framework

The ISV Licensing framework is new to Microsoft Dynamics AX 2012 and enables ISVs to prevent unauthorized access to their solution.   ISVs have been empowered to generate their own license keys and provide those keys to their customers for authorized usage of their solution. This requirement is mandatory so that customers have a consistent licensing procurement experience from Certified for Microsoft Dynamics ISVs.

Requirements:

- Confirm that all entry points in the ISV solution have been secured to the ISV's license key. That is, license code is tied to a configuration key and the configuration key is tied to an entry point.
- Confirm that the ISV license code has the ISV's x509 1024/2048 bit Authenticode certificate assigned to it.

3.2.4   Verification steps

The ISV has provided the required documentation.

3.2.5   Results

- ❑ Pass   Documentation is provided and reviewed. Specific statement of compliance with the foundation framework guidelines is received.
- ❑ Fail   ISV does not certify compliance with the foundation framework guidelines.
- ❑ Fail   ISV documentation deliverables are incomplete or contain errors.

3.2.6   Resources

Documentation and white papers:

- Migrating extended data type relations in Microsoft Dynamics AX 2012
- Services and AIF
- Help System
- Eventing
- ISV Licensing Framework
- Security Framework
    - o Security for Microsoft Dynamics AX (MSDN)
    - o Setting up and maintaining security (TechNet)
- Workflow
    - o Implementing Workflow for Microsoft Dynamics AX (MSDN)

- o [Workflow (TechNet)](#)
- [Analytics](#)
- [Enterprise Portal (EP) and Role Centers (TechNet](#))
- Model Based Development and Deployment
  - o [Deploying customizations and solutions by using models and the model store (TechNet)](#)
  - o [Models](#)
- SQL Server Reporting Services
  - o [Reporting](#)

3.3 Document Compliance with the Microsoft Dynamics AX Application Functional Extensions

3.3.1 Type of Requirement: AOT

3.3.2 Purpose

Correct usage of the Microsoft Dynamics AX application functional extensions is critical to ensuring the integrity of the application, providing better maintainability and supportablity, and to providing a consistent experience for the end user.

Best Practice checks are used to ensure basic uptake of frameworks. In addition to passing Best Practice checks, we require that ISVs be aware of the following guidelines sign off that their applications have correctly followed them. By signing off, an ISV is certifying that they understand proper usage of the frameworks and have have implemented their solutions using the frameworks properly.

3.3.3 Documentation requirements

3.3.3.1 Human Resources framework

In Microsoft Dynamics AX 2012, the Human Resources (HR) framework has been enhanced to provide greater flexibility and sustainability across the organization. The data model has also been completely revised to permit many of the core HR tables to be shared and to enable new functionality. Therefore, developers will need to update every reference made to human resources data entities to reference the new tables.

Requirements:
- Provide a list of all (both new and extended) HR framework patterns in the solution.
- Provide confirmation that all existing patterns have been converted to the Microsoft Dynamics AX 2012 patterns and all new patterns have been implemented as per the whitepaper - [Implementing and Updating the Human Resources Framework for Microsoft Dynamics AX 2012 Applications](#).
- Provide confirmation that the solution does NOT use the "employee" or "worker" tables outside of the prescribed framework.

14

3.3.3.2   Organization model

The organization model in Microsoft Dynamics AX 2012 supports creating hierarchical structures that enable a typical cycle of performance improvement that an organization goes through. Scenarios such as types of organizations, organization hiearchies and transactions and data sharing between companies have been enabled in Microsoft Dynamics AX 2012.

Requirements:

- Provide confirmation that any extension of the organization model in the ISV solution has been done using prescribed best practices.
- Provide confirmation that the solution does NOT enable the usage of virtual company functionality.

3.3.3.3   Number Sequence framework

Microsoft Dynamics AX employs a number sequence framework to generate alphanumeric number sequences. The primary purpose of the number sequence framework is to provide unique, user-friendly identifiers while maintaining a continuous or non-continuous alphanumeric sequence.

Requirements:

- Provide a list of all new number sequences in the solution.
- Provide confirmation that any creation of number sequences in the ISV solution has been done using prescribed best practices.
- Provide confirmation that the solution does NOT contain direct foreign key references to the NumberSequence table.

3.3.3.4   Global Address Book framework

The Global address book is a shared repository of people and organizations, internal or external to the organization. A party, which is an individual record in the global address book, can play multiple roles in a single legal entity or across multiple legal entities. The normalization or centralization of the address information is system-wide and not just specific to the global address book. Transactions that include address information no longer store that data in the transaction tables. Instead a reference or pointer to the address record is stored. This includes the source document of the transaction and any journals or packing slips that are generated from the source document. The other group of entities that use the centralized address table are non-party entities such as sites, warehouses, bank accounts, recruitment media, and sales tax authorities.

Requirements:

- Provide confirmation that any uptake or extension of the Global Address Book framework has been done using prescribed best practices as outlined in the white paper [Implementing the address book framework for Microsoft Dynamics AX 2012 applications](#)
- Provide confirmation that any new or existing person or organization type entities use the Global Address Book framework.

3.3.3.5   Financial Frameworks

The account and financial dimensions framework has been enhanced in Microsoft Dynamics AX 2012 to provide substantially more functionality. This has led to changes in the data model. This would require developers to update existing references to ledger accounts and financial dimensions.

Requirement: Provide confirmation that all existing patterns have been converted to the Microsoft Dynamics AX 2012 patterns and all new patterns have been implemented as per the whitepaper - [Implementing the Account and Financial Dimensions Framework for Microsoft Dynamics AX 2012 Applications](#)

3.3.4   Verification steps

- The ISV has provided the required documentation.

3.3.5   Results

❑   Pass   Documentation is provided and reviewed. Specific statement of compliance with the application framework guidelines is received.

❑   Fail   ISV does not certify compliance with the application framework guidelines.

❑   Fail   ISV documentation deliverables are incomplete or contain errors.

3.3.6   Resources

White Papers:

- [Implementing and Updating the Human Resources Framework](#)

- [Implementing and Extending the Organization Model](#)

- [Using the Enhanced Number Sequence Framework](#)

- [Implementing the Address Book Framework for Microsoft Dynamics AX 2012 Applications](#)

- [Shared Currencies and Exchange Rates for Microsoft Dynamics AX 2012](#)

- [Implementing the Account and Financial Dimensions Framework](#)

4 External Certification

4.1   Managed code solutions must comply with the Microsoft Windows User Experience Guidelines, or its own established guidelines

4.1.1   Type of Requirement: Managed Code

4.1.2   Purpose

The goal for any solution is to establish a high quality and consistency baseline for the user experience in your application.

4.1.3   Verification steps

- For .NET Framework-based solutions, the ISV must certify compliance with a set of user experience guidelines, either the published Microsoft Windows User Experience Interaction Guidelines or its own set of user experience guidelines. If using a non-Microsoft set of published guidelines, the guidelines must be included in the documentation submitted to the tester.
- The tester will verify compliance by sample testing various forms within the ISV solution.

4.1.4   Results

- ❑   Pass   The ISV solution is .NET-based and adheres to the designated user interface guidelines.
- ❑   Fail   The ISV solution is .NET-based and does not adhere to the designated user interface guidelines.

4.1.5   Resources

For information, see the Windows User Experience Interaction Guidelines:
http://msdn.microsoft.com/en-us/library/aa511258.aspx

4.2   MorphX-based solutions must comply with the Microsoft Dynamics AX User Experience Guidelines

4.2.1   Type of Requirement: AOT

4.2.2   Purpose

The user interface design of a solution must provide a user experience that is consistent with the aesthetics of Microsoft Dynamics AX by following the user experience (UX) guidelines established for this release. This includes adherance to the guidelines for:

- MorphX based forms
- Enterprise Portal forms
- Reports
- Role Centers

In rare cases, a departure from the user experience requirements may be necessary due to functional requirements, performance requirements or specific legal requirements or documents. In this case, request an exception in your submission package by including a description of the deviation and describing why the exception is necessary. Requests will be reviewed by Microsoft.

4.2.3   Verification steps

Use the appropriate content to evaluate the ISV solution against the Microsoft Dynamics AX User Experience Guidelines.

- Run the Best Practice tool for Forms – Forms Style with warnings enabled on the ISV solution to ensure that all new forms tied to a FormStyle are in compliance. Please note that all warnings OUTSIDE the 1400 ID range should be ignored.

- The tester will verify compliance by sample testing various forms within the ISV solution.

4.2.4   Results
- ❑   Pass   The ISV solution adheres to Microsoft Dynamics AX User Experience Guidelines.
- ❑   Fail   The ISV solution does not adhere to Microsoft Dynamics AX User Experience Guidelines, or has exceptions that have not been approved by Microsoft.

4.2.5   Resources
For information, see the following:

- Microsoft Dynamics User Experience Guidelines (http://msdn.microsoft.com/en-us/library/gg886610(AX.60).aspx)
- Microsoft Dynamics AX Developer Guidelines (http://msdn.microsoft.com/en-us/library/gg880996(v=AX.60).aspx)

4.3   A solution must be compiled on .NET Framework 4.0 or 4.0 SP1

4.3.1   Type of Requirement: Managed Code

4.3.2   Purpose
ISV solutions must use the same version of Microsoft .NET Framework as the Microsoft Dynamics AX development team uses.

4.3.3   Verification steps
- With the solution running on a test machine, a tester uses Clrver.exe to validate that the ISV running processes are compiled on .NET Framework 4.0 or later.

4.3.4   Results
- ❑   Pass   The ISV solution was compiled on the correct .NET version.
- ❑   Fail   The ISV solution was not compiled on .NET Framework 4.0 or later.

4.3.5   Resources

4.3.6   Resources
For more information, see the following:

- .NET Framework 4.0 - http://msdn.microsoft.com/en-us/library/w0x726c2.aspx
- Clrver.exe (CLR Version Tool) - http://msdn.microsoft.com/en-us/library/ff427522.aspx

4.4 A solution must pass FxCop best practice tests

4.4.1 Type of Requirement: Managed Code

4.4.2 Purpose

FxCop is a code analysis tool that checks managed code assemblies for conformance to the Microsoft .NET Framework design guidelines. FxCop uses reflection, MSIL parsing, and call graph analysis to inspect assemblies for more than 200 defects. It is encouraged that an ISV uses the Visual Studio 2010 Code Analysis tools for early discovery of these defects during design time and FxCop as preparation for the certification test.

FxCop includes the following rule libraries, based on the .NET Framework design guidelines that are loaded by default when a new project is created:

- Design: Rules that detect potential design flaws. These coding errors typically do not impact the execution of your code.
- Globalization: Rules that detect missing or incorrect usage of information related to globalization and localization.
- Interoperability: Rules that support interaction with COM clients.
- Mobility: Rules that support efficient power usage.
- Naming: Rules that support adherence to the naming conventions of the .NET Framework Design Guidelines.
- Performance: Rules that support high-performance libraries and applications.
- Portability: Rules that support portability across different platforms.
- Security: Rules that support safer libraries and applications.
- Security Transparency:
- Usage: Rules that support appropriate usage of the .NET Framework..

Rules are assigned one of five importance levels:

- Critical error: Issues that are highly visible, that prevent code from operating correctly in common scenarios, or both. Critical error messages should be resolved first, and should be excluded only after carefully assessing the impact of ignoring the error.
- Error: Issues at this level have less impact on usability and behavior than critical errors, but should not be excluded without careful assessment.
- Critical Warning: Issues that typically have little or no negative impact on code behavior; they are primarily concerned with code maintainability and correcting less-than-optimal choices for visible elements. However, for a minority of cases, these messages are considered errors and so they should be reviewed closely before being excluded.
- Warning: Issues that are typically concerned with doing things correctly to keep your code base stable, extensible, and maintainable.

- Informational: Messages returned by rules that report information about a target, rather than detecting errors in a target.

4.4.3   Verification steps

1. Download FxCop V10.0 and evaluate all managed code assemblies in the ISV solution.

2. Review rules violations with a level of "Critical Error" or "Error".

3. For any rules violations with a level of "Critical Error" or "Error", or any rules violations excluded in source, the ISV must provide justification.

4.4.4   Results

❑   Pass   The ISV solution does not have any rules violations with a level of "Critical Error" or "Error" reported from FxCop, or a valid justification has been provided for each "Critical Error" or "Error".

❑   Fail   The ISV solution contains "Critical Error" or "Error" rules violations, without justification.

4.4.5   Resources

For more information, see the following:

- Analyzing Managed Code Quality by Using Code Analysis http://msdn.microsoft.com/en-us/library/dd264939.aspx
- FxCop Web site http://msdn.microsoft.com/en-us/library/bb429476(VS.80).aspx
- Download FxCop v10.0 http://www.microsoft.com/downloads/en/details.aspx?FamilyID=917023F6-D5B7-41BB-BBC0-411A7D66CF3C

4.5   A solution must not produce Best Practice Tool errors

4.5.1   Type of Requirement: AOT

4.5.2   Purpose

When using the Microsoft Dynamics AX development environment, you should adhere to the set of best practices that the Best Practice tool checks for. The tool performs a quantitative test on the solution, and logs any errors or warnings that it identifies. These issues can result in best practice errors, warnings, or informational messages.

ISVs should run the Microsoft Dynamics AX Best Practice Tool on their solution periodically during the development and test phases of the product.

Note: Do not use the comment function to avoid running the tool on portions of the code. The only exception to this rule is related to security errors. See the Microsoft Dynamics AX Writing Secure X++ Code white paper for details.

4.5.3   Verification steps

1. Run the Best Practice tool on the ISV solution with the Warning level set to "Errors only".

2. All errors must be fixed or have a written justification included in the submission package and must be reviewed by Microsoft. Exceptions can be granted by Microsoft.

4.5.4 Results
❑ Pass   There were no errors reported by the Best Practice tool, or errors have a written justification and have been reviewed and an exception granted by Microsoft.
❑ Fail   There were errors reported by the Best Practice tool that do not have an exception granted by Microsoft. Resources

4.5.5 Resources
See the following topics in the [Best Practices for Microsoft Dynamics AX Development](#) content on MSDN:

- New Best Practices
- Solving Errors
- Microsoft Dynamics AX Design Patterns
- Using the InfoLog System

4.6 A solution must have an About window

4.6.1 Type of Requirement: AOT

4.6.2 Purpose
In order to adequately identify the solutions installed with Microsoft Dynamics AX 2012, each solution must show up in the Show installed models in the Microsoft Dynamics AX About window.

4.6.3 Verification steps
1. The ISV solution shows up in the Models installed window when the user clicks on the Show installed models and displays the following information:

  - The name of the solution and description (Model name and Model description)
  - Version and build information for the solution
  - Name of the ISV (Model publisher)
  - Signed = Yes

4.6.4 Results
❑ **Pass**   The ISV solution shows up in the Models installed window and displays information correctly
❑ **Fail**   The ISV solution does not show up in the Models installed window
❑ **Fail**   The ISV solution does not display accurate and complete information as described above

4.7 A solution must not use extended data types to define entity relationships

4.7.1 Type of Requirement: AOT

4.7.2 Purpose

Extended Data Types (EDTs) were used in Microsoft Dynamics AX 2009 to define entity relationships. With the new relational enhancements in Microsoft Dynamics AX 2012 a developer should no longer use EDTs to define entity relationships.

4.7.3 Verification steps

- Run the EDT relation Migration tool located in **Tools** > **Code upgrade** > **EDT relation migration tool** to verify that the ISV solution does not use EDTs to define entity relationships.

4.7.4 Results

❑ Pass ISV Solution has defined all Entity relationships properly and the EDT relation migration tool does not find instances of where EDTs have been used to define entity relationships.

❑ Fail The EDT relation migration tool finds instances where EDTs have been used to define entity relationships.

4.7.5 Resources

White paper: [Migrating extended data type relations in Microsoft Dynamics AX 2012)](#)

4.8 A solution must use the Microsoft Dynamics AX 2012 help system to deliver documentation

4.8.1 Type of Requirement: General, AOT

4.8.2 Purpose

The Microsoft Dynamics AX help system supports creating and publishing custom help documentation. The help system includes a server and client:

- The Help Server uses a centralized Web service to respond to requests for help information from the client and stores the Help documentation
- The Help Viewer is an application that is deployed with the Microsoft Dynamics AX client and displays the Help documentation for the client.

Documentation for a Microsoft Dynamics AX ISV solution must provide a user experience that is consistent, both in terms of writing style and depth of information, with the base documentation provided with Microsoft Dynamics AX.

You should add new topics to describe a new process, form, class, or other component that your solution adds to Microsoft Dynamics AX. To provide the documentation, create and publish one or more content elements for the new topic.

You can update an existing topic to document changes, customizations, or enhancements that your solution makes to an existing process, form, class, or other component in Microsoft Dynamics AX. An update requires that the content element is added to the existing content elements that other publishers have for that topic. Other content elements from other publishers may be hidden if the customization has made other content and functionality obsolete.

Identify and justify any content that is being blocked.

4.8.3   Verification steps
- Verify that the ISV solution's user documentation is integrated with the Microsoft Dynamics AX Help System.
- Verify that the ISV solution's form-level user documentation is available via F1.
- Verify that filtering on the ISV content works properly, via the Help View Options button
- Verify, if applicable, where the ISV soluition has added to the content of an existing help topic, validating that it displays in addition to the existing topic.
- Verify, if applicable, where the ISV solution has blocked Microsoft content, validating the blocked content is not necessary for the general usage of the form.

4.8.4   Results
- ❑ Pass   The ISV solution's user documentation is integrated with the Microsoft Dynamics AX 2012 Help system.
- ❑ Pass   The ISV solution's form-level Help is available from the F1 key.
- ❑ Pass   The ISV solution's user documentation can be filtered via the Help View Options button
- ❑ Pass   The ISV solution's additions to existing help content follows guidelines for adding content and/or blocking content.
- ❑ Fail   The ISV solution does not use the Microsoft Dynamics AX 2012 Help system for its user documentation.
- ❑ Fail   The ISV solution's Help is not available from the F1 key
- ❑ Fail   The ISV solution's Help cannot be filtered via the Help View Options button
- ❑ Fail   The ISV solution's additions to existing help content disrupts the existing topics and blocks critical information needed by the user to use the form

4.8.5 Resources

For comprehensive information on adding, updating, and publishing Help topics, see the Help System topic on MSDN and in the Developer Documentation available from the F1 key in the Developer Workspace.

- The Writing Help section of the documentation describes how to create Help documents that Microsoft Dynamics AX Help system can display. The following processes are discussed:

  - Using a template to create a document that adds or updates a topic in Microsoft Dynamics AX Help
  - Using the metadata properties of the template to specify how the Help Server and Help Viewer display the document
  - Using a metadata property of the template to identify the Help document that is displayed when the user presses F1 from a Microsoft Dynamics AX form or other workspace item
  - Creating an XML file that contains metadata to add a new entry to the table of contents in the Help Viewer

- The Publishing Help section of the documentation describes how to add Help documents and table of contents entries to Microsoft Dynamics AX Help system. The following processes are discussed:

  - Publishing new or updated content elements
  - Publishing table of content entries
  - Adding a document set to the Application Object Tree
  - Using common troubleshooting steps to determine why published content elements or table of contents entries do not appear in the Help Viewer

- HTML and Microsoft Word templates are available for creating topics that follow the same general flow/structure as the rest of the topics in the Microsoft Dynamics AX 2012 Help system

4.9 A solution must be localizable / translation

4.9.1 Type of Requirement: General

4.9.2 Purpose

Localization is the practice of adapting digital content or solutions from the locale they were created in to other locales for the sake of increasing profits from sale. This typically involves translation of text strings, but can also include other changes to the software to respect locales, customs, taboos, and other cultural factors.

A solution that is localizable has been developed in a way so that it can be efficiently localized without any changes to the source code.

For each locale where the ISV solution will be sold, you should provide:

- An overview of how you plan to localize the application.
- Evidence that strings are separated from code.
- Translated label files.

4.9.3 Verification steps

1. Verify that the ISV has provided a localization plan for each locale where the ISV solution will be sold.

2. Verify that the source code does not contain hard-coded strings

   - X++-based code – use the Best Practice tool (see Requirement 1.6)
   - .NET Framework-based code – the ISV can either provide a document describing the localization technique used to ensure that there are no hard-coded strings in their solution or provide an output from the Code Analysis Tool for Microsoft Visual Studio 2010 (run the tool with "Microsoft Globalization Rules" enabled).

3. Verify that the ISV has provided translated label files for each locale where the ISV solution will be sold.

4.9.4 Results

- ❑ Pass   The ISV provided a localization plan for each locale where the ISV solution will be sold.
- ❑ Pass   The ISV solution does not contain hard-coded strings.
- ❑ Pass   The ISV provided translated label files for each locale where the ISV solution will be sold.
- ❑ Fail   The ISV did not provide a localization plan or translated label files for each locale, or the ISV solution contains hard-coded strings.
- ❑ N/A   The ISV solution is not sold outside of US-English locales.

4.9.5 Resources

Use specialized tools that recycle translations of repeated text and resize application user interface elements to allow for localized text and graphics.

Microsoft Dynamics AX has a built-in label system that lets you identify the labels and F1 help text in the ISV application. This system simplifies translation of the application to other languages. Also, the Microsoft Dynamics AX Best Practices tool can determine whether there are any hard-coded strings in your application code.

- Localization Planning
- Best practices for labels

For Managed Code solutions, the Code Analysis Tool (only available with Visual Studio Premium and Ultimate editions) has a rule set called Microsoft Globalization Rules which will return errors if there are hardcoded strings in the code.

4.10 A solution must follow globalization rules

4.10.1   Type of Requirement: General

4.10.2   Purpose
Globalization is the process of developing solutions or writing content whose features and characteristics are language, culture, and market neutral. The ISV solution must operate correctly in all locales (user and system settings for language, culture, country/region, and related data formatting). The code will support any combination of editing languages, user locales, or user interface languages with a single worldwide binary.

4.10.3   Verification steps
The ISV solution must handle date and times, currency, and string formats correctly regardless of the language or locale of:

- Microsoft Dynamics AX
- Windows
- SQL Server

Note: For .NET Framework-based applications, the FxCop tool is also used to check for compliance as part of Requirement 4.4.

4.10.4   Results
❑   Pass   The ISV solution correctly displays strings, date/time, and currency when Microsoft Dynamics AX is installed with a different locale.
❑   Pass   The ISV solution correctly displays strings, date/time, and currency when the Windows operating system is using a different locale.
❑   Pass   The ISV solution correctly displays strings, date/time, and currency when the SQL Server database is using a different locale.
❑   Fail   The ISV solution does not correctly display strings, date/time, or currency when Microsoft Dynamics AX is installed with a different locale.
❑   Fail   The ISV solution does not correctly display strings, date/time, or currency when the Windows operating system is using a different locale.
❑   Fail   The ISV solution does not correctly display strings, date/time, and currency when the SQL Server database is using a different locale.

4.10.5   Resources
To simplify the globalization of X++-based applications, Microsoft Dynamics AX has built-in data types for time and dates. Additionally, Microsoft Dynamics AX has a built-in label system that ensures that the labels and help text can be easily changed for use with different languages.

The .NET Framework provides built-in support for globalization. It is fully Unicode-enabled and provides extensive built-in globalization support. Microsoft Dynamics AX is Unicode-enabled as of version 4.0.

26

For more information, see the following:

- [Microsoft Go Global Developer Center](http://msdn.microsoft.com/en-us/goglobal/) (http://msdn.microsoft.com/en-us/goglobal/)
- [FxCop Globalization Warnings](http://msdn.microsoft.com/en-us/library/ms182184.aspx) (http://msdn.microsoft.com/en-us/library/ms182184.aspx)

4.11 A solution must support the same infrastructure as Microsoft Dynamics AX

4.11.1 Type of Requirement: General

4.11.2 Purpose

In order to provide a seamless experience for our partners and customers, the ISV solution must support the same infrastructure versions specified by Microsoft Dynamics AX and cannot require any version of a technology that conflicts with those required by Microsoft Dynamics AX.

It is possible that an ISV solution might require some technology that is not mentioned in the Microsoft Dynamics AX platform matrix list for example, an ISV solution requiring Microsoft Kinect as the user interface device.

4.11.3 Verification steps

Verify that the ISV solution installs and runs successfully on the infrastructure version(s) required by Microsoft Dynamics AX (browser, database, operating system, and other required software).

See the matrix of supported versions in Appendix C: Platform Testing Matrix

4.11.4 Results

❑ Pass   The ISV solution installs and runs successfully on the infrastructure version(s) specified in Appendix C.

❑ Fail   The ISV solution does not install or run successfully on the infrastructure version(s) specified in Appendix C.

❑ Fail   The ISV solution requires a version of browser, database, operating system, or other software that is in conflict with the version required by Microsoft Dynamics AX.

4.11.5 Resources

For more information see Microsoft Dynamics AX 2012 Deployment (http://technet.microsoft.com/en-us/library/gg731870(AX.60).aspx)

4.12 The ISV Must Provide Database Upgrade Scripts and use the Microsoft Dynamics AX Upgrade Framework

4.12.1   Type of Requirement: AOT

4.12.2   Purpose

The architecture of an ISV solution should integrate smoothly with the upgrade framework if the ISV solution existed before Microsoft Dynamics AX 2012. This means that ISVs need to account for all upgrade scenarios and, usually, provide upgrade scripts that will upgrade all of the customer databases related to ISV features.

The new source-to-target upgrade model supports several kinds of upgrade scripts appropriate for various scenarios:

- Preprocessing scripts or preprocessing forms - Customized scripts that run during preprocessing, and forms to provide them with a user interface. These are used when upgrading the ISV solution requires substantial and time-consuming data upgrade, or when changes to the data model involve conversion from company specfic tables (DataArea based) to shared tables (non-dataAreaID-based).
- Upgrade readiness scripts - Scripts that identify scenarios which may causedataupgrade scripts to fail or may result in incomplete data upgrade in supported scenarios.
- Postsynch scripts - Similar to postsynch scripts in Microsoft Dynamics AX 2009. Now, however, the ISV should use the new upgrade attribute framework.

The ISV should provide the complete upgrade story, considering the following:

- Look into all upgrade scenarios and in most cases be able to upgrade all customer databases related to the ISV feature.
- For cases where automated upgrade scripts do not exist (which should be the exception), document the mitigations that are in place to allow the customer to plan ahead. For example, there could be an upgrade readiness script to identify a case in which an X++ report is not automatically converted to a SQL Server Reporting Services report.
- Service Pack upgrade for any ISV update should be accounted for.
- Is there complete upgrade documentation, including documentation for any upgrade scripts? Ideally this documentation should be integrated with the Microsoft Dynamics AX help files so that customer has single source for help.

4.12.3   Verification steps

1. An upgrade test should be performed using the ISVs demo data, that is, a base Microsoft Dynamics AX version + ISV solution should be upgraded to Microsoft Dynamics AX 2012. The ISV should document the post upgrade data validation performed to verify the correctness of the upgrade.

2. (Optional)The ISV should also take a customer database through the upgrade process and document the results.

3. The ISV should provide upgrade documentation, consisting of the upgrade procedure and the upgrade script.

4. The ISV should use the upgrade whitepaper to ensure that the upgrade framework is being used properly

   a. The ISV solution should have the following class -ReleaseUpdateDB60_<ISVSolutionName> - where <ISVsolutionName> is the name of the ISV solution

   b. Upgrade methods within the class mentioned above should use upgrade attributes to define upgrade scripts.

4.12.4   Results
❑ Pass   The ISV has provided upgrade documentation and the documented results of their upgrade testing process (on demo data or customer data) for their solution. The ISV solution has the appropriate class and the correct upgrade attributes in methods as described above.
❑ Fail   The ISV was unable to provide either of the documents mentioned above
❑ Fail   The ISV solution does not follow the guidelines provided in the upgrade whitepaper

4.12.5   Resources
Upgrade scripts (http://go.microsoft.com/fwlink/?LinkId=215589)

Upgrade scripts guide (http://go.microsoft.com/fwlink/?LinkId=215587)

CERTIFIED FOR MICROSOFT DYNAMICS - SOFTWARE SOLUTION TEST REQUIREMENTS

4.13 The Installation Procedure Must Be Compatible with Microsoft Dynamics AX

4.13.1   Type of Requirement: General, AOT

4.13.2   Purpose
To ensure that ISV solutions on Microsoft Dynamics AX 2012 follow a consistent installation methodology approved by Microsoft.

Microsoft recommends the use of the Windows Installer, an application installation and configuration service. The Windows Installer is an operating system component that centrally manages application installation configuration and application uninstall. The Windows Installer lets the operating system manage application setup and configuration, and provides the following benefits:

- It manages reference counting and version checking of shared components.
- It provides a reliable and complete uninstall procedure that includes correct handling of shared components.
- It automatically repairs damaged applications when they are started.
- It uses transaction-based installation: the Windows Installer can roll back to the earlier installed version of the application without error.

4.13.3   Verification steps
1. AOT based solutions must be shipped as a signed model file.

2. AOT based solutions' Help content must be packaged as an MSI and discovery of the existing Microsoft Dynamics AX 2012 Help Content must be automated.

3. All of the above artifacts must be packaged together as an MSI (optional).

4. An installation guide that explains the install process must be provided.

5. A tester will use the installation guide to install the ISV solution and confirm that the procedure is consistent with the approved guidelines.

4.13.4   Results
❑   Pass   The ISV has provided an install guide which follows the Microsoft Dynamics AX 2012 approved installation methodology.
❑   Pass   A managed code ISV solution uses Windows Installer
❑   Fail   The ISV did not provide an install guide or the install guide does not follow the Microsoft approved installation methodology for Microsoft Dynamics AX 2012 ISV solutions.
❑   Fail   Managed code ISV solution does not use Windows Installer
❑   Fail   The ISV solution is shipped as an unsigned model file.

4.14 Functional Tests must be executed and pass for the ISV solution

4.14.1 Type of Requirement: AOT, Managed Code

4.14.2 Purpose

To increase the quality of solutions that run in the Microsoft Dynamics AX environment and give the market the assurance that ISV solutions built on Microsoft Dynamics AX meet technical requirements that ensure a high standard. An essential aspect in support of this goal is ensuring that the ISV solution functions as expected after it is installed.

Functional test scenarios must be mapped to the high-level features of the ISV solution to demonstrate the breadth of the coverage.

4.14.3 Verification steps

1. Review the mapping of functional test scenarios to high-level features of the ISV solution provided in the documentation requirement section of this test.

2. Execute the functional tests provided by the ISV to verify end-to-end scenarios for the solution.

4.14.4 Results

❑ Pass   A mapping of functional test scenarios to high-level features was provided.

❑ Pass   All functional tests had actual results that matched expected results (test passed) and there were no catastrophic failures noted (i.e., system crash).

❑ Fail   One or more of the functional tests had actual results that did not match expected results (test failed) or there was a catastrophic failure noted (i.e., system crash).

4.14.5 Resources

Guidance for creating and submitting functional tests for your solution is found in Appendix A: Functional Testing Guidance.

For more information, see the following:

- Microsoft Dynamics AX Testing Guidance White Paper
- Dave Froslie's blog
- Visual Studio Team Test blog
- Visual Studio Lab Management blog

4.15 A solution must include sample data

4.15.1   Type of Requirement: General

4.15.2   Purpose
Sample data, also known as demonstration or demo data, is useful for many purposes, such as sales demonstrations, training, and testing. The standard Microsoft Dynamics AX database includes demo data.

ISVs are required to deliver at least one set of sample data with their solution with instructions that describe how to add the demonstration data to Microsoft Dynamics AX.

4.15.3   Verification steps
1.   Follow the instructions supplied by the ISV to install the sample data and verify that it was installed successfully.

2.   If sample data is installed as part of the core install process, verify that there is a mechanism for excluding the data from the install process.

4.15.4   Results
❑   Pass   Instructions for installing sample data were provided and by following the steps, the data was installed and accessible.
❑   Pass   Installing sample data is part of the core installation of the ISV's solution and a mechanism was available to exclude the data from the install process.
❑   Fail   Instructions for installing sample data were not provided, or instructions were provided but the data could not be installed or accessed.
❑   Fail   Installing sample data is part of the core installation of the ISV's solution, but there is no mechanism for excluding the data as part of the install process.

4.15.5   Resources
You must supply instructions that describe how to add the sample data to Microsoft Dynamics AX.

Some data centers do not permit installing sample data on production servers. For this reason, you can deliver sample data as part of the main installation or as a separate installation (for example, VPC). If you include the sample data as part of the core installation, you must provide a mechanism (such as a checkbox) to exclude the data from the install process.

For more information, see the Microsoft Dynamics AX 2012 Demo Tools and Materials (requires a PartnerSource log in)

4.16 Back Up and Restore the Solution and Data

4.16.1 Type of Requirement: General

4.16.2 Purpose

A customer or partner must be able to back up and restore a solution and all associated data.

For an X++ application, new fields and tables will be treated the same as any other data and will be backed up as part of the core Microsoft Dynamics AX backup process. For data that is stored in a separate database, you must provide backup and restore procedures.

If your solution does not have a unique backup or restore procedure, it is sufficient to refer to the standard Microsoft Dynamics AX procedures.

4.16.3 Verification steps

Execute the backup and restore procedures provided for the ISV's solution and data.

4.16.4 Results
- ☐ Pass   Instructions for performing a backup and restore of the ISV's solution and data have been provided (or referenced) and are successful.
- ☐ Fail   Instructions for performing a backup and restore of the ISV's solution and data have not been provided (or referenced), or the backup or restore failed.

4.16.5 Resources

For more information, see [Introduction to Backup and Restore Strategies in SQL Server](http://msdn.microsoft.com/en-us/library/ms191239.aspx) (http://msdn.microsoft.com/en-us/library/ms191239.aspx).

4.17 The ISV must establish and follow the Microsoft Security Development Lifecycle best practices

4.17.1 Type of Requirement: General

4.17.2 Purpose

The SDL is a software development security assurance process that consists of a collection of security practices, grouped by the phases of the traditional software development life cycle. Combining a holistic and practical approach, the SDL introduces security and privacy throughout all phases of the development process, with the goal of optimizing your software security and helping to protect end users. Practical experience at Microsoft has shown that security activities executed in chronological order and as part of a repeatable process result in greater security gains and cost benefits than those resulting from ad hoc implementation.

- To ensure the authenticity and integrity of the ISV solution to customers and partners, code must be digitally signed. Digital signatures placed on files by code signing help provide a foundation for security by enabling downstream users or processes to identify the origin of the software and to verify whether the software has been tampered with post-release.
- For security reasons, managed assemblies must have a strong-name signature applied. StrongName is a technology used to assign an identity to an assembly. Strong-name signing:
  - Gives assemblies unique identities.
  - Prevents a third party from modifying code without the CLR noticing.

4.17.3   Verification steps

Microsoft's SDL website provides detailed guidance on the best practices and associated tools

- Training Requirements

- Security Requirements

- Quality Gates/ Bug Bars

- Security and Privacy Risk Assessment

- Design Requirements

- Attack Surface Reduction

- Threat Modeling

- Use Approved Tools

- Deprecate Unsafe Functions

- Static Analysis

- Dynamic Program Analysis

- Fuzz Testing

- Threat model and attack surface review

- Incident Response plan

- Final Security Review

- Release/Archive

These practices have also been described in detail in the Simplified Implementation of the Microsoft SDL white paper. The ISV must use all applicable tools as described here. The table below shows the tools that will be run on the ISV solution during the certification testing process.

Technology	Tool	Details	Comments
Managed Code	FxCop	FxCop is a separate requirement, but the	FxCop should not

		ISV should still ensure all security errors are addressed per this requirement	generate any errors
Managed Code	Microsoft Code Analysis Tool	In the command window, change to the "CAT.NET" installed directory; type 'catnetcmd /file:"file name"'.	Tool should not report any problems.
All files and controls	SignTool	Run the SignTool verify command with the "/a" (all) option against the ISV's code files, including any third-party code used by the solution.	All files and controls in the ISV solution, including any third-party files or controls used by the solution, have been digitally code signed.
All Binaries	Binscope Binary Analyzer	Launch BinScope.exe, in the Configure tab, select file name on Target, and select to run all Checks	BinScope should not report any failed checks
Managed Code	Strong Name Tool	Run Sn.exe with the "-vf" (forced verify) option against the ISV's .NET assemblies, including any third-party .NET assemblies used by the solution	All managed assemblies in the ISV solution have a strong-name signature applied.

4.17.4   Results
❑ Pass   No critical errors are generated (see table above) when applicable SDL tools are run on the ISV solution.
❑ Fail   One or more applicable SDL tools generate critical errors when run on the ISV solution.
Note: Not all SDL development security requirements apply to every ISV solution.

4.17.5   Resources

[SDL website](#)
[SDL process](#)
[SDL tools](#)

Microsoft Authenticode Technology provides the ability to digitally sign files. For more information, see [Authenticode](#) ([http://msdn.microsoft.com/en-us/library/ms537359(VS.85).aspx](#)).

The Sn.exe ([http://msdn.microsoft.com/en-us/library/k5b5tt23(v=VS.100).aspx](#)) tool provided with the Microsoft Visual Studio .NET development system supports the proper use of strong-names. For more information see [Creating and Using Strong-Named Assemblies](#) in the .NET Framework 4 Programming with Assemblies section on MSDN.

5 APPENDIX A: MICROSOFT DYNAMICS KEY USAGE SCENARIO TEMPLATES

Company:

Solution:

Version:

Prepared by:

Date Prepared:

☐ Identify the intended purpose of the product.

Specify what fundamental service that the product is supposed to provide. To the extent possible, define the audience for the product. Write a paragraph that briefly explains the purpose of the product and describes its intended audience.

Example:

Test Solution is a solution for managing data warehouse that enables the user to define the data warehouse strategy in a multifacility and multiuser environment.

Describe the product purpose:

☐ Identify the most representative roles for your solution.

Define which functional work roles you're designing your product for and what role centers represent each role.

Example:

Project Manager (Role Center ID = 9051): A Project Manager will typically be responsible for Project Initiation, Scheduling, Resource Management and Project Budget management.

☐ Identify the most representative primary functions for each role of your solution.

Term	Definition	Notes
Primary function	Main usage of the product — the function that is so important that, in the estimation of a typical user, its inoperability or impairment would	A function is *primary* if you can associate it with the purpose of the product and it is essential to that purpose. Primary functions define the product. For example, the function of adding text to a document in Microsoft Office Word is so important that the product would be

	make the product unfit for its purpose.	useless without it. Groups of functions that are taken together could be a primary function. For example, although no single function on the drawing toolbar of Word is primary, the complete toolbar might be primary. If the toolbar is primary, most functions on that toolbar should be operable for the product to pass the test.

Examples of primary functions:

Manage cross-docking operations.

Manage stock environment.

Manage IT interoperability with a fast carrier company.

List all primary functions:

☐   Identify the most representative contributing functions for each role.

Term	Definition	Notes
Contributing function	Any function that contributes to the utility of the product but is not a primary function.	Although contributing functions are not primary, their inoperability could be a reason for failure. For example, users may be able to do useful things with a product, even if it has an Undo function that never works, but most users will find that unacceptable. Such a failure would violate fundamental expectations about how products should work.

Example of a contributing function:

Generate a 3-D report.

List all contributing functions:

☐ Specify potential instabilities and challenging data.

   List five to 10 functions or groups of functions (preferably primary functions) for focused instability testing.

   Specify challenging data for each selected function. Think of large, complex, or otherwise challenging input.

Examples:

Functions that interoperate with other products (for example, object linking and embedding, file conversion).

Functions that handle events external to the application (for example, wake up a sleeping computer when a fax arrives).

Functions that make intensive use of memory.

Functions that interact extensively with the operating system.

Functions of unusual complexity.

Functions that change operating parameters (for example, preference settings).

Functions that manipulate operating system configuration.

40

Microsoft Dynamics is a line of integrated, adaptable business management solutions that enables you and your people to make business decisions with greater confidence. Microsoft Dynamics works like and with familiar Microsoft software, automating and streamlining financial, customer relationship and supply chain processes in a way that helps you drive business success.

U.S. and Canada Toll Free 1-888-477-7989

Worldwide +1-701-281-6500

www.microsoft.com/dynamics

Microsoft