

Dynamics AX

Microsoft® Dynamics™ AX 2009

Microsoft Dynamics ISV Software Solution Test Guidelines



The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This test specification is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© (2007) Microsoft Corporation. All rights reserved.

Microsoft, Microsoft Dynamics, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

Introduction	6
This Document.....	6
Program Overview.....	6
Contents of the Test Guidelines.....	6
Product Versions Supported	7
Types of Solutions	7
Retesting	8
More Information	8
Testing Process	9
Checklists	10
First-Time Software Test Requirements.....	10
Software Retest Requirements	12
ISV Software Solution Requirements and Recommendations	13
Development Requirements.....	13
1.1 A .NET Framework–based ISV Application Must Be Compiled on .NET Framework 2.0 or Later and Must Pass the Required FxCop Tests.....	14
1.2 An ISV Application Must Not Produce Best Practice Tool Errors.....	15
1.3 An ISV Application Must Have an About Window	17
1.4 An ISV Application Should Follow Microsoft Dynamics AX Architectural Guidelines	18
1.5 Managed Assemblies Must Be Strong Named	19
1.6 ActiveX Controls Should Be Digitally Signed.....	19
1.7 Application Integration Framework (AIF) Data Object Must Be Synchronized with the Artifacts Used to Define It.....	20
1.8 Workflow Templates Should Adhere to Best Practices and Use AX Infrastructure	21
User Assistance and Documentation Requirements.....	22
2.1 The ISV Must Provide an Implementation Guide.....	22
2.2 ISV Application Online Documentation Must Use the Microsoft Dynamics AX Help Navigation Structure.....	23
2.3 ISV Application Online Documentation Should Follow the Microsoft Help Style Guidelines.....	24
User Experience Requirements.....	25
3.1 An ISV Application Must Comply with Microsoft Dynamics AX User Experience Guidelines	25
Reporting Requirements.....	26
4.1 An X++ Application Should Follow Microsoft Dynamics AX Reporting Guidelines; .NET Framework–based ISV Applications Should Document the Reporting Guidelines Used	26
4.2 An ISV Application Should Follow the SQL Reporting Services Implementation Guidelines.....	27
Translation and Localization.....	28
5.1 An ISV Application Must Follow Globalization Rules	28

5.2 An ISV Application Must Separate Strings from Source Code	29
Technology, Configuration, and Platform Requirements	30
6.1 An ISV Application Must Support the Infrastructure that Microsoft Dynamics AX Supports	30
Setup Requirements	30
7.1 The ISV Application Installation Procedure Must Be Compatible with Microsoft Dynamics AX.....	31
7.2 The ISV Application Must Correctly Register DLLs and ActiveX Controls	32
7.3 The ISV Application Setup Program Must Verify That the Correct Software Versions Are Installed	33
7.4 The ISV Application Must Verify That Required Microsoft Dynamics AX Modules Are Installed	33
7.5 The ISV Setup Program Must Inherit or Create Configuration Key Settings.....	34
7.6 The ISV Application Must Include Installable Demonstration Data.....	35
Backup and Restore	36
8.1 The ISV Must Include Procedures to Back Up and Restore the Application and the Data	36
Extensibility and Customization Requirements	37
9.1 (X++) The ISV Must Document How to Customize the Application.....	37
9.2 (.NET Framework) The ISV Must Document How to Customize the Application.....	38
Upgrade and Maintenance	38
10.1 The ISV Must Document All Integration Points	39
10.2 The ISV Must Provide Database Upgrade Scripts	39
10.3 The ISV Must Use File Versioning for DLLs and ActiveX Controls	40
Sustainability.....	41
11.1 The ISV Must Remove Non-Functioning Code from Code Base.....	41
Trustworthy Computing.....	42
12.1 The ISV Must Establish and Follow Secure Development Best Practices.....	42
12.2 Before Development Begins, ISV Staff Should Complete Security and SDL Training.....	44
Appendix A: User Assistance UI Requirements.....	46
Procedure Help Requirements	46
Form Help Requirements.....	46
Short Help Requirements.....	47
Appendix B: User Experience Guidelines	48
Requirements for Application Windows.....	48
Navigation Layer.....	48
Navigation Pane Requirements.....	48
Area Page Requirements.....	49
List Page Requirements	49
Task Forms Requirements	49
Dialog Boxes	51
Wizard Requirements.....	52
Message Boxes	52
Controls	54
Commands	56

Requirements for Enterprise Portal	57
EP Page Types	57
Role Center Controls	60
EP Controls	63
EP Naming	69
EP Navigation	69
Requirements for Reports	70
SRS Report – Full Page Requirements	70
Requirements for Icons and Symbols	71
Appendix C: Microsoft Dynamics AX Consistency Verification Test	72

Introduction

This Document

The Microsoft® Dynamics™ AX ISV Software Solution Test Guidelines describes the technical requirements that an application must meet to integrate and operate with Microsoft Dynamics AX 2009. For information about testing of the requirements, see the "How to Comply" and "Test Methodology" sections in each requirement description.

Program Overview

Welcome to the ISV Software Solution Test Guidelines for Microsoft Dynamics AX 2009. This document describes the requirements that an Independent Software Vendor (ISV) application must meet to integrate and operate with Microsoft Dynamics AX.

The goal of the test is to increase the quality of applications that run in the Microsoft Dynamics AX environment. The purpose of this test is to give the market the assurance that ISV applications built on Microsoft Dynamics meet technical requirements that ensure a high standard.

The test guidelines are designed to walk you through the test process and to help you make sure that your application meets the requirements.

This document provides the following:

- An explanation of the testing process
- Definitions of the software requirements
- Descriptions of development, test, and documentation best practices

To pass the test, an ISV must demonstrate the development quality of its product and its ability as a software company to maintain and enhance that product in the future. Therefore, the test includes a technical review and an in-lab inspection. The actual test is administered and conducted by a third-party vendor.

We welcome your comments and suggestions. Please contact dyncert@microsoft.com with your feedback.

Contents of the Test Guidelines

The Microsoft Dynamics AX ISV Software Solution Test Guidelines describes the test requirements, recommendations, and best practices. The guidelines are presented in individual, subject-based modules, some of which may be common to other Microsoft Dynamics tests.

This document contains the following sections:

- Introduction (this section) explains the purpose and high-level requirements of the test.
- [Testing Process](#) describes how the testing process works, from qualification through communication of test results.
- [Checklists](#) provides a summary of the documentation that you must submit with your application.

- [ISV Software Solution Requirements and Recommendations](#) defines in detail each requirement category, how these requirements are tested, and what you can do to make sure that your application meets the requirements.

Product Versions Supported

The intent of the test is to support Microsoft’s latest shipping version of a product. For that reason, ISV applications submitted for testing must run on Microsoft Dynamics AX 2009 with the latest service pack installed.

Types of Solutions

Microsoft Dynamics solutions fall into three general categories and three setup complexity levels. The category and setup complexity of a solution determines (in part) the type and complexity of the testing requirements and the costs associated with testing the solution.

Figure 1 shows the different solution categories and setup complexity levels.

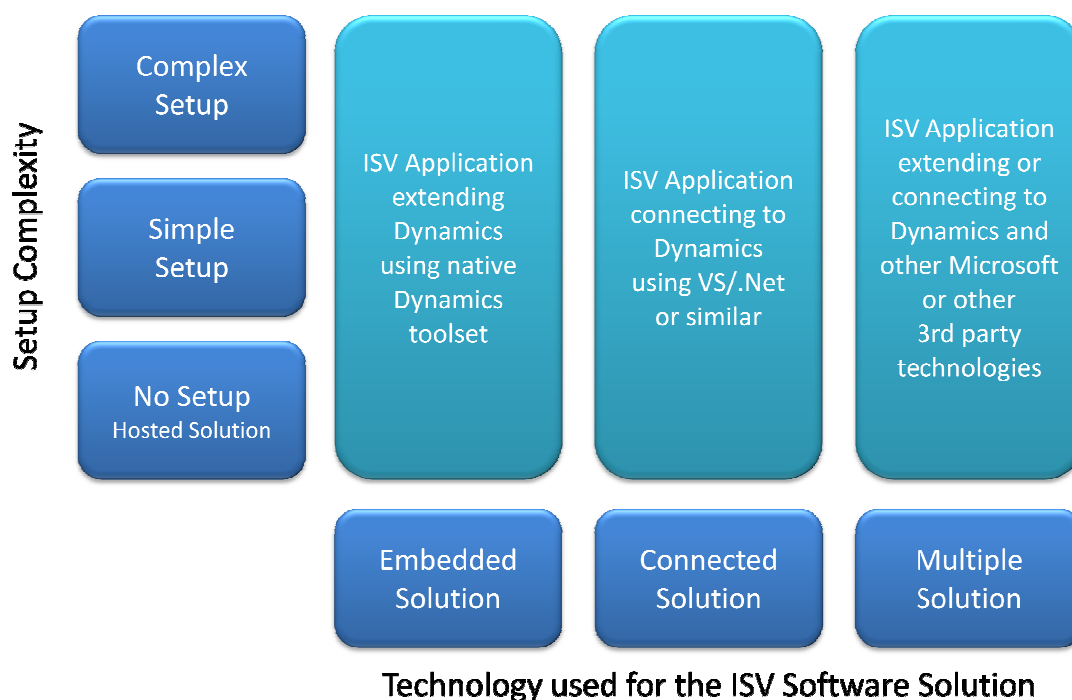


Figure 1
Solution categories and solution setup complexity levels

ISV software solution complexity falls into one of the following categories (listed from least complex to most complex):

- An embedded or *in-product* solution is an ISV application that extends Microsoft Dynamics by using only the tools provided with the Microsoft Dynamics product. For example, an embedded solution can be built in a proprietary development environment, such as the Microsoft Dynamics NAV C/SIDE, C/AL environment, or the Microsoft Dynamics AX Morph, X++ environment.
- A *connected* solution is an ISV application that uses Microsoft Visual Studio, the .NET Framework, or similar tools to connect to the Microsoft Dynamics product.

Typically, a connected solution refers to a standalone product that interoperates with the core Microsoft Dynamics product by using it as a business rules engine. The solution might establish interoperability by using Web Services, .NET Framework assemblies, COM interop, or some other means. The solution might or might not be .NET Framework–based; however, it must run on a Microsoft operating system.

- A *multiple* solution is one that extends or connects to Microsoft Dynamics and other Microsoft or other third-party technologies.

Setup complexity falls into one of the following categories (listed from least complex to most complex):

- No setup (potentially a *hosted* solution) provides services to customers, who do not have to purchase, install, or maintain the software or hardware. Hosted solutions have no setup requirements for end users; however, installing and configuring a hosted solution can be extremely complex, and the test vendor may not have the hardware, custom software, or services that the solution requires.
- A *simple* setup is one that the test vendor can install and configure without requiring a restorable backup, virtual PC (VPC), or other additional assistance.
- A *complex* setup is one that the test vendor cannot completely replicate; for example, solutions that require specific hardware, custom software, or back-end services that the vendor cannot duplicate.

Retesting

Test results are valid for 24 months. When retested, the ISV application must be updated to support the latest Microsoft Dynamics version, including the latest service pack.

More Information

For more information about the functionality of Microsoft Dynamics AX, see the Microsoft Dynamics AX Home Page at <http://www.microsoft.com/dynamics/ax>.

For more information about the Microsoft Partner Program, see the Microsoft Worldwide Partner Portal Home Page at <http://partner.microsoft.com>.

For more information about how the ISV test helps you earn partner program points, see the ISV Software Testing Framework page at <http://partner.microsoft.com/global/program/competencies/40011374>.

For more information about the Microsoft Dynamics ISV/Software Solutions competency, see the Microsoft Dynamics Testing for ISVs page at <http://partner.microsoft.com/global/program/competencies/40013116>.

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 referenced on the Microsoft Web page. The vendor site contains a description of the test, as well as an application form with a published test fee schedule.

Depending on the type of solution (embedded, connected, or multiple) and the solution setup (simple or complex), different test methods will apply for which the test fee will vary. You can make your solution available to the test vendor for testing by using any of the following methods:

- Providing the software with installation instructions for the test vendor to install
- Sending a virtual server image of a working configuration of the product to the test vendor
- Using an interactive Live Meeting session to provide access to a working configuration of the product

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.
- You must submit a number of documents as part of the test. These are identified in the appropriate test modules, as well as in a summary checklist. See the [Checklists](#) section of this document.
- You must upload documents and your solution to the test vendor's servers for testing. If your setup is complex, you must be prepared to use Microsoft Live Meeting to demonstrate the solution to the test vendor.

Checklists

The checklists in this section describe the required items that you must include when you submit your software solution for testing or retesting.

First-Time Software Test Requirements

The following checklist describes the documentation that you must include with your solution. Note that a single document may contain information that satisfies multiple requirements. For that reason, the actual number of documents might be significantly smaller than the number of items in this list. Please use the "Check" column to indicate either:

- The filename of the document containing the required information
- "X" if the requirement has been satisfied and does not require additional documentation

Requirement	Check
The ISV solution (your application software) and product documentation. This can be in the form of a CD or other distributable media, or you can use a virtual server image or Live Meeting session to demonstrate your software.	
Describe the business functionality that the ISV solution provides and examples of key usage scenarios (see Appendix C, Consistency Verification Test).	
Describe and justify any FxCop warnings in the Security category. See Requirement 1.1 .	
Describe and justify any exceptions to Best Practice rules. See Requirement 1.2 .	
The ISV solution must have an About window. See Requirement 1.3 .	
List of vendor or third-party assemblies that are excluded from the strong name requirement. See Requirement 1.5 .	
Describe and justify any exceptions to AIF Best Practice rules. See Requirement 1.7 .	
Implementation guide appropriate for VARs or others who intend to deploy your solution. See Requirement 2.1 .	
The ISV solution must have online help in the .chm file format and Help topics must be accessible from the F1 function key. See Requirement 2.2 .	
Describe and justify any exceptions to user experience requirements. See Requirement 3.1 .	
Describe and justify any FxCop warnings in the Globalization category. See Requirement 5.1 .	
Describe and justify any exceptions to hard-coded string Best Practice rules. See Requirement 5.2 .	

The ISV solution must run on the specified infrastructure versions that Microsoft Dynamics AX runs on. See Requirement 6.1 .	
List of all resources that the ISV solution adds to Microsoft Dynamics AX and complete instructions for uninstalling the solution. See Requirement 7.1 .	
List of all registry settings generated during installation. See Requirement 7.2 .	
Document the name of the method added to the ApplicationVersion class and the version number of the ISV solution. See Requirement 7.3 .	
List of the Microsoft Dynamics AX components that are required by the ISV solution. See Requirement 7.4 .	
The ISV solution must inherit standard Microsoft Dynamics AX configuration settings, or create new settings. See Requirement 7.5 .	
Sample data for testing and instructions for loading or installing it. See Requirement 7.6 .	
Describe backup and restore procedures. See Requirement 8.1 .	
Customization and extensibility guide that documents how to extend your solution (this is commonly known as a developers guide). See Extensibility and Customization Requirements 9.1 or 9.2.	
Document the integration points of the ISV solution. See Requirement 10.1 .	
DLLs and ActiveX controls must be file versioned. See Requirement 10.3 .	
Describe and justify any exceptions to non-functioning code Best Practice rules. See Requirement 11.1 .	
Secure development best practices must be established and followed. See Requirement 12.1 .	

Software Retest Requirements

An ISV must submit an application for retesting in the following cases:

- Whenever a new major version of Microsoft Dynamics AX is released, including SP releases containing major new functionality
- Whenever a major version of the ISV application is released, including SP releases containing major new functionality

The following checklist describes the documentation that you must include with your application when you submit it for retesting. Note that a single document might contain information that meets multiple requirements. For that reason, the actual number of documents could be significantly smaller than the number of items in this list. Please use the "Check" column to indicate either:

- The filename of the document containing the required information
- "X" if the requirement has been satisfied and does not require additional documentation

Requirement	Check
The ISV application (your application software) and product documentation. This can be in the form of a CD or other distributable media, or you can use a virtual server image or Live Meeting session to demonstrate your software.	
Describe the changes to objects and components.	
Describe the changes to the data model.	
Upgrade scripts for the new release. See Requirement 10.2 .	
<i>What's New</i> document that describes the business functionality of the new release.	
Updated documentation for any requirement that may have changed since the solution was submitted for the First-Time Software Test.	

ISV Software Solution Requirements and Recommendations

The Microsoft Dynamics AX ISV solution test requirements ensure that ISV applications integrate with Microsoft Dynamics AX without causing system problems or errors. Microsoft and third-party test vendors worked together to define the minimum requirements that an ISV application must meet to operate successfully in a Microsoft Dynamics AX environment.

Note: The test does not validate the correctness or relevance of ISV application functionality.

This section describes the test requirements and recommendations and the procedures for verifying that each requirement is met. In this document, the word *must* in the text of a requirement means that the item or feature is not optional. The word *should* means that the item or feature is recommended and its inclusion is a best practice, but it is not strictly required. However, these recommendations will be considered for inclusion as requirements in later versions of this test.

Some requirements are technology-specific and do not apply to all ISV applications. Therefore, each requirement indicates the type of ISV technology to which it applies. Additionally, an ISV application might include several technologies. In these situations, the vendor will test those parts of the application that use the technologies that the requirement or recommendation applies to.

If a requirement is defined as applicable to X++, it applies to either of the following:

- Any code written in X++ (either business logic or code that implements an integration to an external component), if the vendor in-lab test is performed directly on the code.
- Any application that includes X++ code, if an in-lab test is not performed directly on the code.

Similarly, if a test is defined as applicable to *External*, it applies to either of the following:

- Any code not written in X++ (including DLLs, ActiveX controls, services, applications that have their own user interface, and so on), if the vendor in-lab test is performed directly on the code.
- Any application that includes such code, if an in-lab test is not performed directly on the code.

Each requirement includes a table that indicates the type of technology (X++ or external) and type of solution setup (simple, complex, and host-based) that the requirement applies to.

Development Requirements

Your application must meet the following development requirements:

- [Requirement 1.1: A .NET Framework–based ISV application must be compiled on .NET Framework 2.0 or later and must pass the required FxCop tests.](#)
- [Requirement 1.2: An X++-based ISV application must not produce best practice tool errors.](#)
- [Requirement 1.3: An ISV application must have an About window.](#)
- [Recommendation 1.4: An ISV application should follow Microsoft Dynamics AX architectural guidelines.](#)
- [Requirement 1.5: Managed assemblies must be strong name.](#)
- [Recommendation 1.6: ActiveX Controls Should be Digitally Signed.](#)

- [Requirement 1.7 Application Integration Framework \(“AIF”\) object needs to be instantiated correct.](#)
- [Requirement 1.8 Workflow Templates Should Adhere to Best Practices and Use AX Infrastructure.](#)

1.1 A .NET Framework–based ISV Application Must Be Compiled on .NET Framework 2.0 or Later and Must Pass the Required FxCop Tests

Type	Test Method	Technology	Solution Category		
Required	In-lab test	External	Simple	Complex	Hosted
		Managed code only	✓	✓	✓

Summary and Intent

Applications must use the latest release of the Microsoft .NET Framework and pass the required FxCop tests. FxCop is a code analysis tool that checks managed code assemblies for conformance to the Microsoft .NET Framework design guidelines.

Note: If the application has a small number of unmanaged code elements these will not have to go through the FxCop test.

Resources

For more information, see the following:

- FxCop Web site: [http://msdn2.microsoft.com/en-us/library/bb429476\(vs.80\).aspx](http://msdn2.microsoft.com/en-us/library/bb429476(vs.80).aspx)
- Download: <http://code.msdn.microsoft.com/codeanalysis/Release/ProjectReleases.aspx?ReleaseId=553>
- .NET Framework Web site: <http://msdn2.microsoft.com/en-us/netframework/default.aspx>

How to Comply

An ISV can download FxCop from the FxCop Web site. FxCop uses reflection, MSIL parsing and call graph analysis to inspect assemblies for more than 200 defects.

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

- **COM:** Rules that detect COM Interop issues.
- **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.
- **Naming:** Rules that detect incorrect casing, cross language keyword collisions, and other issues related to the names of types, members, parameters, namespaces, and assemblies.
- **Performance:** Rules that detect elements in your assemblies that will degrade performance.

- **Security:** Rules that detect programming elements that leave your assemblies vulnerable to malicious users or code.
- **Usage:** Rules that detect potential flaws in your assemblies that can impact code execution.

Rules are assigned one of five importance levels:

- **Critical error:** Issues that is 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.

No errors should be reported in any rules in the security category (exceptions can be granted for warnings). And all security exceptions must be reviewed by Microsoft.

Test Methodology

The test vendor will execute the FxCop analysis on the ISV application, using the FxCop version 1.35. No errors should be reported in any rules in the security category (exceptions can be granted for warnings). All security exceptions must be reviewed by Microsoft.

Criteria for Passing

This requirement is mandatory. If the ISV application does not pass this requirement, it will fail the test.

1.2 An ISV Application Must Not Produce Best Practice Tool Errors

Type	Test Method	Technology	Application Category		
Required	In-lab test	X++	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

ISV X++ applications must use the same coding standards that the Microsoft Dynamics AX development team uses. Microsoft Dynamics AX has a built-in tool that checks whether an ISV application follows Microsoft Dynamics AX best practices. The tool performs a quantitative test on the application, and logs any errors or warnings that it identifies. Your application must not produce errors when this tool is used. And you should run through the list of warnings to make sure that best practice is implemented.

Resources

For more information, see the following topics in the "[Best Practices for Microsoft Dynamics AX Development](#)" section of Microsoft Dynamics AX Help:

- New Best Practices: <http://msdn2.microsoft.com/en-us/library/aa887087.aspx>
- Solving Errors: <http://msdn2.microsoft.com/en-us/library/bb530207.aspx>
- Microsoft Dynamics AX Design Patterns
- Using the InfoLog System

How to Comply

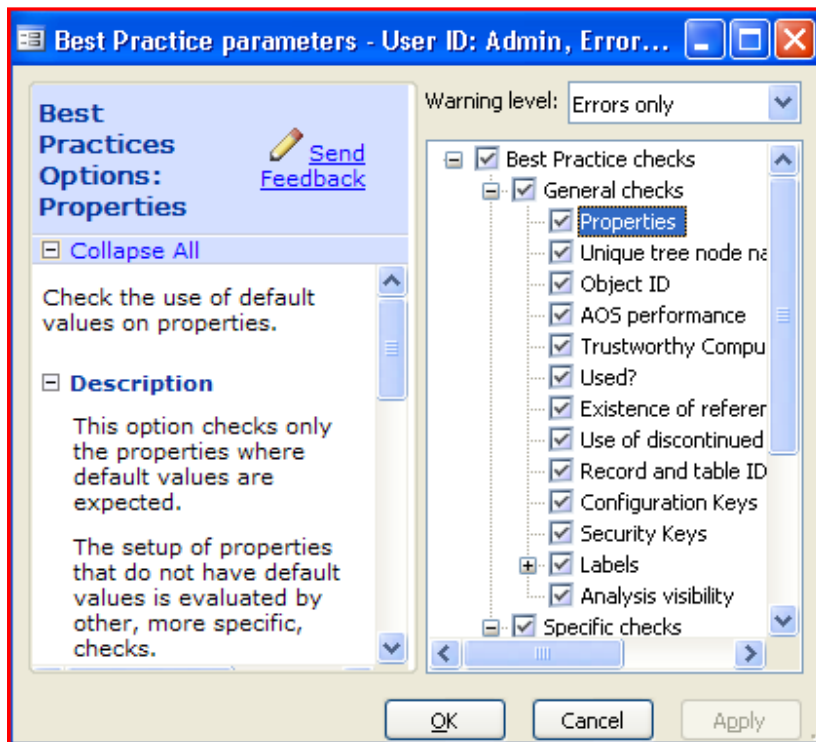
For X++ applications, refer to the Microsoft Dynamics AX Development Best Practices section of Microsoft Dynamics AX Help before and during your development and test processes. It is easier to avoid problems rather than to fix them. Periodically, run the Microsoft Dynamics AX Best Practices Tool on your application. Track and fix any errors (bugs) that the tool identifies. Be sure that the tool produces no errors before you submit your application for testing. If you receive best practices errors and you cannot fix them, provide a written justification for allowing the errors, and include it as part of your submission package. Include any justifications in your checkpoint document.

Do not use the comment function to avoid running the tool on portions of your code. The only exception to this rule is related to security errors. Refer to the [Microsoft Dynamics AX Writing Secure X++ Code](#) white paper for details.

Note: The presence of warnings in the Best Practice tool event log will not cause your application to fail this requirement. However, you should review all warnings carefully, examine the appropriate code, and document the results of your inspection. Use the checkpoint form to document your findings.

Test Methodology

For X++ code, the test vendor will run the Best Practices tool with the warning level "Errors only" (see picture below) and will then review the Microsoft Dynamics AX event log to determine if the application produced any errors. If the application has a large number of errors that are impractical to fix prior to the audit, the ISV must produce a plan to remedy the problems before the application is released.



Criteria for Passing

This requirement is mandatory. If the application produces errors or becomes unstable, it will fail the test.

1.3 An ISV Application Must Have an About Window

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

ISV applications must have an About window that displays the name of the application, contact details for the ISV, and version and build information for the application.

ISVs must extend the existing ApplicationVersion class of Dynamics to add the information about the application and version number. For related information, see [section 7.3](#).

Resources

For more information, refer to the *Microsoft Dynamics AX Developers Guide*, located on the Microsoft Dynamics AX product CD.

How to Comply

The AX Help menu is in managed code, so it is not possible to add a menu item to it. However, the Dynamics AX About window can be modified by adding a new method to the ApplicationVersion class and then overriding the buildNo method. Add a button to the AX About window (the label should be the ISV application name) that pops up an About Window for your solution.

Test Methodology

The test vendor will verify there is an About window that is accessible from the Dynamics AX About window, and that it extends the existing ApplicationVersion class from Dynamics AX.

Criteria for Passing

This requirement is mandatory. If the ISV application does not include an About window, it will fail the test.

1.4 An ISV Application Should Follow Microsoft Dynamics AX Architectural Guidelines

Type	Test Method	Technology	Solution Category		
Recommendation	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Microsoft Dynamics AX Help has sections that describe design concepts and standards. You should follow these guidelines to make sure that your application complies with Microsoft Dynamics AX architectural best practices and design patterns.

Resources

For more information, see the following topics in Microsoft Dynamics AX Help:

- Designing a Microsoft Dynamics AX Application
- Microsoft Dynamics AX Design Patterns
- Introduction to the Microsoft Dynamics AX Business Connector (instructions for creating custom Help).

In addition, refer to the MSDN topic, Developing for Microsoft Dynamics AX, available at: <http://msdn.microsoft.com/en-us/library/aa493467.aspx>.

How to Comply

When you design your application, you should document your design documents in specifications. You must include these documents in your test submission package. Conduct design reviews to make sure that your application uses the design patterns and document your reviews in review reports.

Test Methodology

The test vendor will review your design documents and code for compliance. The test vendor will review a representative sample of application objects to make sure that they comply with Microsoft Dynamics AX

guidelines and with the design pattern documentation you submit. The vendor will pay particular attention to tables, forms, classes, and query objects to make sure that they conform to established guidelines.

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

1.5 Managed Assemblies Must Be Strong Named

Type	Test Method	Technology	Solution Category		
Required	In-lab review	External	Simple	Complex	Hosted
		Managed Code	✓	✓	✓

Summary and Intent

This requirement is included for security purposes.

Resources

The Sn.exe tool provided with the Microsoft Visual Studio® .NET development system supports the proper use of strong names.

How to Comply

The ISV must apply strong naming to managed assemblies. The exception is that if the ISV application uses a vendor or third-party assembly, the assembly does not need to be signed. The ISV should provide a list of vendor or third-party assemblies.

Test Methodology

The test vendor will use the Sn.exe tool provided with Visual Studio .NET to verify the proper use of strong names.

Criteria for Passing

This requirement is mandatory. If the ISV does not use strong naming for managed assemblies, the application will fail the test.

1.6 ActiveX Controls Should Be Digitally Signed

Type	Test Method	Technology	Solution Category		
Recommended	In-lab review	External	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

This recommendation is included for security purposes. Digital signing helps users decide if they want to trust a control and assures users that files have not been tampered with.

Resources

Code signing certificates are available from several vendors, as described at <http://msdn2.microsoft.com/en-us/library/ms995347.aspx>.

The Microsoft Windows SDK Sign Tool is available on MSDN at: <http://msdn.microsoft.com/en-us/library/8s9b9yaz.aspx>.

How to Comply

After the ISV obtains a code signing certificate, the ISV should use the Microsoft Windows SDK Sign Tool to sign the files. The exception is that if the ISV application uses a vendor or third-party assembly or ActiveX control, the control does not need to be signed. The ISV should provide a list of vendor or third-party controls.

Test Methodology

During testing, the test vendor will note any warnings about ActiveX controls without valid certificates.

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

1.7 Application Integration Framework (AIF) Data Object Must Be Synchronized with the Artifacts Used to Define It

Type	Test Method	Technology	Solution Category		
Required	In-lab review	External	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

When using the Application Integration Framework you must ensure that your object is synchronized correctly with the underlying artifact.

Resources

The Best Practice tool has a check to see if there is missing or extra methods in the underlying data object. For more information, see the developer help topic at: Help > Developer Help > Microsoft Dynamics AX SDK > Integrating Other Applications > Application Integration Framework Overview > Best Practices: Application Integration Framework.

How to Comply

Run the Best Practice tool. For more information on running the tool see [section 1.2](#).

Test Methodology

During testing, the test vendor will note any best practice errors.

Criteria for Passing

This requirement is mandatory. If the ISV best practice tool produce errors, the application will fail the test.

1.8 Workflow Templates Should Adhere to Best Practices and Use AX Infrastructure

Type	Test Method	Technology	Solution Category		
Recommended	In-lab review	All	Simple	Complex	Hosted
	✓		✓	✓	✓

Summary and Intent

Microsoft Dynamics AX provides a workflow infrastructure that you can use when developing customizations or modules. These requirements are intended to provide the AX users with a consistent experience from a UI and workflow functionality perspective.

Resources

The Best Practice tool has checks that focus on the properties of workflow categories, templates, approvals, and tasks. For more information see the developer help topic at: Help > Developer Help > Microsoft Dynamics AX SDK > Workflow > Best Practices: Workflow.

For information about how to develop workflow templates please consult the developer help topics at: Help > Developer Help > Microsoft Dynamics AX SDK > Workflow.

How to Comply

Run the Best Practice tool. For more information on running the tool see [section 1.2](#).

For forms that are to be used to interact with workflow, use the "WorkflowEnabled" property on form design to enable the workflow UI controls. In addition, the canSubmitToWorkflow override method must be implemented.

Test Methodology

During testing, the test vendor will note any best practice errors.

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

User Assistance and Documentation Requirements

Your application must/should comply with the following user assistance and documentation requirements:

- [Requirement 2.1: The ISV must provide an implementation guide for the application.](#)
- [Requirement 2.2: Online documentation for an ISV application must use the Microsoft Dynamics AX Help navigation structure.](#)
- [Recommendation 2.3: Online documentation for an ISV application should follow Microsoft Dynamics AX Help style guidelines.](#)

2.1 The ISV Must Provide an Implementation Guide

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

You must include a partner-facing implementation guide in your documentation.

ISV partners and customers who use or deploy a Microsoft Dynamics AX application must be able to successfully deploy, configure, and manage the application in an existing Microsoft Dynamics AX environment. Your documentation must provide information that allows your partners and customers to successfully install or upgrade your application in such an environment.

Resources

For more information, see the [Microsoft Dynamics AX Implementation Guide](#).

How to Comply

To meet this requirement, you must include adequate system requirements, installation, configuration, and upgrade information to allow a partner to implement your application in a new or existing Microsoft Dynamics AX environment.

If you don't have unique hardware or system requirements, it is sufficient to reference Microsoft Dynamics AX requirements.

If yours is a hosted solution, the documentation you submit for this requirement should explain that there is no implementation guide provided for your solution and why.

Note: Providing a security hardening guide is a trustworthy computing (security) requirement. You can meet the hardening guide requirement by including a section in your implementation guide that describes how to deploy your application in a secure manner.

Test Methodology

The test vendor will review your documentation to verify that you have included adequate implementation information.

A satisfactory guide contains the following sections:

- Description of the solution (the problem it solves)
- Hardware environment
- Operating system requirements
- Installation checklist
- Installation guide
- Operational checklist (daily, monthly, and annual procedures as well as how to perform backups, and so on)

Criteria for Passing

This requirement is mandatory. If the ISV application documentation does not include an implementation guide, the application will fail the test.

2.2 ISV Application Online Documentation Must Use the Microsoft Dynamics AX Help Navigation Structure

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Your documentation must be easy for the user to access and to navigate. Documentation for a Microsoft Dynamics AX ISV application should provide a user experience that is consistent with the base documentation provided with Microsoft Dynamics AX. Therefore, your documentation must be in the Help .chm file format and must integrate with the Microsoft Dynamics AX Help system for X++ applications. Users must be able to see Help topics by pressing the F1 function key, which starts the Microsoft Dynamics AX Help window. For .NET Framework–based applications, please visit the MSDN HTML Help link in the Resources section, below.

Resources

For more information, see the following:

- For specific information about how user assistance and Help information should appear in your application’s user interface, see the [User Assistance UI Requirements](#) section in this document.
- The [Help Kit for Microsoft Dynamics AX](#) contains the tools and files for creating Help file topics that have the same look and feel as Help topics in Microsoft Dynamics AX. The Help Kit shows you how to connect and integrate your topics to the existing Help in Microsoft Dynamics AX.

- For information about Help for .NET Framework–based applications, see the MSDN HTML Help guidelines at [http://msdn2.microsoft.com/en-us/library/aa189109\(office.10\).aspx](http://msdn2.microsoft.com/en-us/library/aa189109(office.10).aspx).

How to Comply

To satisfy this requirement, use the documents listed in the Resources section, above, to design and create your Help system.

Test Methodology

The test vendor will review your Help documentation for compliance and usability. The test vendor will review a representative sample of application modules to make sure that Help is available by pressing F1 and that it follows the navigation structure of the core Microsoft Dynamics AX Help system.

Criteria for Passing

This requirement is mandatory. If the ISV application online documentation does not follow the structure of the core Microsoft Dynamics AX Help system, the application will fail the test.

2.3 ISV Application Online Documentation Should Follow the Microsoft Help Style Guidelines

Type	Test Method	Technology	Solution Category		
Recommended	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

The online documentation for your application should follow Microsoft Help style guidelines. It should tell the user how to perform specific tasks, and it should be easy for the user to understand. Documentation for a Microsoft Dynamics AX ISV application should 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.

Resources

For more information, see the MSDN online Help style guidelines at <http://msdn2.microsoft.com/en-us/library/ms997600.aspx>.

How to Comply

Make sure that you have online help and that it provides meaningful information. The guidelines on MSDN will help you to create appropriate content.

Test Methodology

The test vendor will review your Help documentation for style, accuracy, and usability. The vendor will review a representative sample of application modules to make sure that Help topics are appropriate, easy to understand, correct, and adhere to style and user interface guidelines.

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

User Experience Requirements

Your application must comply with the mandatory user experience guidelines and it is recommended that you also comply with the recommended guidelines:

- [Requirement 3.1: An ISV application must comply with mandatory Microsoft Dynamics AX User Experience guidelines.](#)

3.1 An ISV Application Must Comply with Microsoft Dynamics AX User Experience Guidelines

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

For X++ code, you should always try to avoid Microsoft Dynamics AX compliance problems rather than trying to correct them. To produce a compliant Microsoft Dynamics AX application, follow these general recommendations:

- Read and understand the Microsoft Dynamics™ User Experience Guidelines for AX 2009, available on the [Downloads tab](#) of the AX Developer Center.
- Use caution if you need to override existing Microsoft Dynamics AX core functionality.
- Run the Best Practice tool and consider the results carefully.
- Become familiar with the Windows User Experience guidelines and the Microsoft Dynamics AX User Assistance Best Practices Handbook.

For .NET Framework–based code, you should identify and adhere to a standard set of UI Guidelines.

Resources

See [Appendix B](#) for User Experience guidelines.

How to Comply

To determine whether your module complies with the User Experience Guidelines, review each item in Appendix B. If you do not have the resources to perform a complete review of your application, you should complete at least two randomly selected checks for each form.

For .NET Framework–based applications, you must provide the user interface guidelines that you are following. The test vendor will sample various UI controls and forms to verify that the guidelines are being followed.

In rare cases, you might have to depart from the user experience requirements because of functional requirements, or because you must adhere to specific legal requirements or match legal documents. If you believe that this is the case for your Microsoft Dynamics AX application, submit a description of the deviation, describing the following:

- Why the exception is necessary.
- What you have done to make sure that you cannot accomplish the overall user goal without violating the Microsoft Dynamics AX user experience requirements.

If your module deviates from user experience recommendations, you must provide a justification for these deviations also. However, the acceptance threshold is somewhat reduced for recommendations.

Test Methodology

The test vendor will review your application to verify that it meets the user experience requirements. The vendor will visually review and use a representative sample of UI elements (windows, forms, wizards, Help, and so on) to make sure that they look and function according to the guidelines.

Criteria for Passing

This requirement is mandatory. If the ISV application does not follow the mandatory user experience guidelines, it will fail the test.

Reporting Requirements

Your application should meet the following reporting recommendations:

- [Recommendation 4.1: An X++-based ISV application should follow the reporting guidelines in Microsoft Dynamics AX Help. A .NET Framework–based ISV application should provide documentation for the reporting guidelines used, and should adhere to them.](#)
- [Recommendation 4.2: SQL reporting services should adhere to recommended implementation for Microsoft Dynamics AX.](#)

Additionally, Microsoft Dynamics AX–based applications should use the tools and templates provided with Microsoft Dynamics AX.

4.1 An X++ Application Should Follow Microsoft Dynamics AX Reporting Guidelines; .NET Framework–based ISV Applications Should Document the Reporting Guidelines Used

Type	Test Method	Technology	Solution Category		
Recommended	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Microsoft Dynamics AX provides a report wizard that you must use when you design reports in Microsoft Dynamics AX. This wizard guides the user through the steps for standard reports.

Resources

A white paper that describes the SQL reporting engine in Microsoft Dynamics AX is available at: <https://mbs.microsoft.com/partnersource/documentation/whitepapers/reportingandbusinessintelligence.htm?printpage=false>.

How to Comply

Use the report template system that is built into Microsoft Dynamics AX for any new standard report that you create. If your application is .NET Framework–based, provide a description of the reporting guidelines you used.

Test Methodology

The test vendor will review your application to verify that it meets Microsoft Dynamics AX reporting requirements. The test vendor will randomly test up to 4 reports on the list to ensure that they are deployed via the wizard, and that the user experience is consistent with other reports delivered in the core product.

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

4.2 An ISV Application Should Follow the SQL Reporting Services Implementation Guidelines

Type	Test Method	Technology	Solution Category		
Recommended	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Microsoft Dynamics AX provides implementation guidelines for SQL reporting in Microsoft Dynamics AX applications. These guidelines ensure that all SQL-based reports are surfaced in a consistent manner and will function accurately in Microsoft Dynamics AX.

Resources

A white paper that describes the SQL reporting engine in Microsoft Dynamics AX is available at: <https://mbs.microsoft.com/partnersource/documentation/whitepapers/reportingandbusinessintelligence.htm?printpage=false>.

Description of the SQL reporting engine working with Microsoft Dynamics AX is available at: <http://www.microsoft.com/dynamics/ax/product/businessintelligencewp.aspx>.

How to Comply

Use the SQL reporting implementation guidelines when you design and implement SQL reports for your Microsoft Dynamics AX ISV applications. Include documentation that verifies that your report covers the guidelines.

Test Methodology

The test vendor will review your application to verify that it meets SQL reporting implementation requirements.

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

Translation and Localization

To simplify globalization, your application must comply with the following requirements:

- [Requirement 5.1: An ISV application must follow globalization rules so that it can run in any country without requiring translation.](#)
- [Requirement 5.2: An ISV application must separate strings \(labels\) from source code.](#)

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

5.1 An ISV Application Must Follow Globalization Rules

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

To simplify 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.

Resources

For more information, see the following:

- [Microsoft Global Development and Computing Portal](#)
- [MSDN Developer Center: Visual Studio Globalization](#)
- [FxCop Globalization Warnings](#)

How to Comply

You must ensure that the application can be setup using local settings for things like time and currency etc.

Test Methodology

During the in-lab test, the test vendor will perform a qualitative review to determine whether your application follows globalization rules with respect to labels and data types for time and dates. Additionally, the vendor will review your Help to make sure that it can be used with different languages.

For .NET Framework–based applications, the FxCop tool is used to check for compliance. For more information on FxCop, see [section 1.1](#).

Criteria for Passing

This requirement is mandatory. If the ISV application does not follow globalization rules, it will fail the test.

5.2 An ISV Application Must Separate Strings from Source Code

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Localization of business applications requires that you implement the correct business processes and practices for a culture/locale. Differences in how cultures/locales conduct business are heavily shaped by governmental and regulatory requirements. Therefore, localization of business logic can be a massive task.

You should use specialized tools that recycle translations of repeated text and resize application UI 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. All label files must be translated into the languages where the ISV application is sold.

Resources

For more information, see the following:

- [MSDN Library: Localization Planning](#)
- Help > Developer Help > Microsoft Dynamics AX SDK > Best Practices > Best Practices for Labels

How to Comply

You must provide an overview of how you plan to localize your application, and you must also provide evidence that strings are separated from code.

Test Methodology

During the in-lab review, the test vendor will run the Best Practice tool to determine if hard-coded strings are in your source code.

Criteria for Passing

This requirement is mandatory. If the ISV application does not separate strings from source code, it will fail the test.

Technology, Configuration, and Platform Requirements

Your application must meet the following technology and platform requirement:

- [Requirement 6.1: An ISV application must support the infrastructure that Microsoft Dynamics AX supports.](#)

6.1 An ISV Application Must Support the Infrastructure that Microsoft Dynamics AX Supports

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Your application must run on the specified infrastructure (browser, database, operating system, and other software) versions that Microsoft Dynamics AX 2009 runs on, or on later versions. Additionally, you must not require any version of a technology that conflicts with those required for Microsoft Dynamics AX.

Resources

For more information see Installing and configuring Microsoft Dynamics AX on MSDN at: <http://msdn2.microsoft.com/en-us/library/aa834394.aspx>.

How to Comply

Test your Setup program on the prescribed infrastructure, and then make sure that your application runs.

Test Methodology

The test vendor will perform a qualitative review to determine whether your application runs on the prescribed architecture (browser, database, operating system, and other required software).

Criteria for Passing

This requirement is mandatory. If the ISV application does not run on the prescribed architecture, it will fail the test.

Setup Requirements

Your application must meet the following installation and removal (uninstall) requirements:

- [Requirement 7.1: The ISV application installation procedure must be compatible with Microsoft Dynamics AX.](#)
- [Requirement 7.2: The ISV application installation procedure must correctly register DLLs and ActiveX controls.](#)

- [Requirement 7.3: The ISV application Setup program must verify that the correct software versions are installed.](#)
- [Requirement 7.4: The ISV application Setup program must verify all required Microsoft Dynamics AX components are installed.](#)
- [Requirement 7.5: The ISV application Setup program must inherit or create configuration settings.](#)
- [Requirement 7.6: The ISV application must include installable demonstration data.](#)

7.1 The ISV Application Installation Procedure Must Be Compatible with Microsoft Dynamics AX

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

You must provide installation instructions, and they must be clear and easy to follow. The installation instructions must include procedures for installing and configuring Microsoft Dynamics AX so that it functions with the ISV add-on application. The instructions should be in the form of a plain text file (they can also be part of the standard user documentation) and must list all necessary steps, including working with the data import and file import, system settings, and instructions for using any automated installation executables.

You must also provide instructions for uninstalling the ISV application, including removal of any imported code, removal of any DLL or ActiveX components, removal of registry entries, and removal of Microsoft Dynamics AX itself. If it is not possible to uninstall the ISV application, you must state this in your documentation. After the application is removed from the system, the Microsoft Dynamics AX installation on the test bed system might not be functional.

Resources

For more information see Installing and configuring Microsoft Dynamics AX on MSDN at: <http://msdn2.microsoft.com/en-us/library/aa834394.aspx>.

How to Comply

For .NET Framework–based applications, 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.

Test Methodology

The test vendor will confirm that the ISV has provided a complete list of all resources added to the Microsoft Dynamics AX application. This list will be used to verify the removal of the product.

- Installation: The test vendor will follow each step in the installation instructions in the order presented. The vendor should be able to complete the installation without consulting support personnel or contacting the ISV.
- Uninstall: The vendor will follow each step in the removal instructions in the order presented, if un-installation is possible. The vendor should be able to remove the product without consulting the ISV. After removal of the application, the Microsoft Dynamics AX application might not be functional.

The test vendor will review the list of components that you provide to verify that the entire ISV product has been removed.

Criteria for Passing

This requirement is mandatory. If the ISV application installation program does not run on Microsoft Dynamics AX, the application will fail the test.

7.2 The ISV Application Must Correctly Register DLLs and ActiveX Controls

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

The Setup program for both Microsoft Dynamics AX-based and .NET Framework-based applications should record any DLLs and ActiveX components in the registry database of the operating system. The registry serves as a central configuration database for user, application, and computer-specific information.

Resources

For information on requirements for how to build DLLs, see: <http://msdn2.microsoft.com/en-us/library/527z7zfs.aspx>.

For information on how to register DLL files see: <http://msdn2.microsoft.com/en-us/library/ms859484.aspx>.

For information on ActiveX controls see: <http://msdn2.microsoft.com/en-us/library/aa751968.aspx>.

How to Comply

Check the registry to make sure that your Setup program functions correctly. Document the correct registry settings, and include this information with your application when you submit it for testing.

Test Methodology

During the in-lab test, the test vendor will install your application and review the registry to verify that the Setup program registers all components.

Criteria for Passing

This requirement is mandatory. If the ISV application installation program does not correctly register DLLs and ActiveX controls, the application will fail the test.

7.3 The ISV Application Setup Program Must Verify That the Correct Software Versions Are Installed

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Microsoft Dynamics AX tracks software versions by adding version numbering to the ApplicationVersion class in the application object tree (AOT). All information about installs upgrade or service pack install are then stored in the Tables SysSetupCompanyLog and SysSetup Log.

Resources

For more information, see the Microsoft Dynamics online Help (press F1) for the AOT elements.

How to Comply

Open the ApplicationVersion class methods, and confirm that a method for the version number of your application has been added. Document the method and version number, and include this information with your application when you submit it for testing.

Test Methodology

The test vendor will perform a qualitative review to determine whether the correct software components are installed.

Criteria for Passing

This requirement is mandatory. If the ISV application installation program does not add the correct version numbering, the application will fail the test.

7.4 The ISV Application Must Verify That Required Microsoft Dynamics AX Modules Are Installed

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

If your application requires certain Microsoft Dynamics AX modules or other applications to be installed, this must be documented and clearly specified in your implementation or setup guide.

If you are able to, you should detect this prior to the install through your setup program. You should perform this check early in the sequence of setup activities so that the installation process can be reversed.

Resources

For more information see Installing and configuring Microsoft Dynamics AX on MSDN at: <http://msdn2.microsoft.com/en-us/library/aa834394.aspx>.

How to Comply

You must provide the test vendor with a list of the Microsoft Dynamics AX components that your application requires. The test vendor will verify different configurations with the required components missing to determine whether the ISV Setup program functions correctly.

Test Methodology

The test vendor will perform a qualitative review to determine whether the correct software components are installed.

Criteria for Passing

This requirement is mandatory. If the ISV application installation program does not install the correct software components, the application will fail the test.

7.5 The ISV Setup Program Must Inherit or Create Configuration Key Settings

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

The application Setup program must have configuration settings that are compatible with the core Microsoft Dynamics AX product. The settings should inherit, not overwrite, standard configuration settings, or the application should create new settings. If the application changes configuration settings, it should display warning messages.

Microsoft Dynamics AX has a built-in system for switching functionality on and off. You use the On and Off configurations keys in the Microsoft Dynamics AX configuration form.

To use this form

1. On the Main menu, point to Administration, point to Setup, and then point to System.
2. Click Configuration.

Resources

See the standard Dynamics AX help by pressing F1 when you are on the Configuration Key form.

How to Comply

Before you submit your application for testing, check the Microsoft Dynamics AX Configuration form to make sure that configuration keys work as required.

Test Methodology

During the in-lab test, the test vendor will review your application to make sure that configuration settings are inherited and that warning messages display for any changed settings.

Criteria for Passing

This requirement is mandatory. If the ISV application installation program does not inherit or create configuration settings, the application will fail the test.

7.6 The ISV Application Must Include Installable Demonstration Data

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Demonstration data is useful for many purposes, such as sales demonstrations, training, and this test. Therefore, ISVs are required to deliver at least one demonstration data with their application. Microsoft recognizes that some data centers do not permit installation of demonstration data on production servers. For this reason, you can deliver demonstration data as part of the main installation or as a separate installation (for example, VPC). If you include the demonstration data as part of the core installation, you must provide a checkbox to exclude the demonstration data from the process.

Resources

For more information see:

<https://mbs.microsoft.com/partnersource/documentation/sampled/ax40demodata.htm?printpage=false>

How to Comply

Use the guidelines in the "Summary and Intent" section, above, to build your demonstration data. Deliver the data with your application as part of the test upload.

The standard Microsoft Dynamics AX database includes demonstration data. Value-added resellers (VARs) use the demonstration data during the sales process to demonstrate the application. They also use the pre-configured data later, during user training.

You must ship the data with the main objects of the application or as separate .dat and .def files.

Additionally, you must supply instructions that describe how to add the demonstration data to Microsoft Dynamics AX.

Test Methodology

To verify this requirement, the test vendor will follow these steps:

1. Compile the objects.
2. Follow the instructions supplied by the ISV to install the demonstration data.

Criteria for Passing

This requirement is mandatory. If the ISV application does not include demonstration data, it will fail the test.

Backup and Restore

Your application must meet the following backup and restore requirement:

- [Requirement 8.1: The ISV must include procedures for backing up and restoring both the application and the data.](#)

8.1 The ISV Must Include Procedures to Back Up and Restore the Application and the Data

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

A customer or partner must be able to back up and restore your application and all associated data. For a Microsoft Dynamics AX–based 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 separate databases (this is most likely to occur in .NET Framework–based applications), you must provide a backup and restore process.

Resources

For more information, see “Upgrading to Dynamics AX” and “Before you Begin Upgrading” at MSDN at: <http://msdn2.microsoft.com/en-us/library/aa497070.aspx>.

How to Comply

For a .NET Framework–based application, you must provide a process for backing up the application itself. If you don’t have unique backup or restore procedures, it is sufficient to reference AX procedures.

Test Methodology

During the in-lab test, the test vendor will verify that you have included backup and restore procedures. The test vendor will perform the backup and restore process to make sure that it functions correctly.

Criteria for Passing

This requirement is mandatory. If the ISV application does not include backup and restore procedures, it will fail the test.

Extensibility and Customization Requirements

Your application must meet the following extensibility and customization requirements:

- [Requirement 9.1: \(X++ applications only\) The ISV must use the AOT to document how to customize the application.](#)
- [Requirement 9.2: \(.NET Framework-based applications\) The ISV must document how to customize the application.](#)

9.1 (X++) The ISV Must Document How to Customize the Application

Type	Test Method	Technology	Solution Category		
Required	In-lab review	X++	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Regardless of the capabilities of your application, partners or customers must be able to extend or customize it to meet their unique requirements. A customer or partner must have the required information to complete these customizations successfully so that the application continues to function as designed and can be upgraded with minimal impact.

Resources

For more information and to see examples and descriptions of design patterns for Microsoft Dynamics AX, go to: <http://msdn2.microsoft.com/en-us/library/ms940503.aspx>

How to Comply

Your documentation must include best practices and guidance on how to correctly customize and extend your application.

Test Methodology

During the in-lab test, the test vendor will verify that you have included customization procedures in Help text and in code comments.

Criteria for Passing

This requirement is mandatory. If the ISV does not document how to customize the application, the application will fail the test.

9.2 (.NET Framework) The ISV Must Document How to Customize the Application

Type	Test Method	Technology	Solution Category		
Required	In-lab review	External	Simple	Complex	Hosted
		Managed code only	✓	✓	✓

Summary and Intent

Regardless of the capabilities of your application, partners or customers must be able to extend or customize it to meet their unique requirements. A customer or partner must have the required information to complete these customizations successfully so that the application continues to function as designed and can be upgraded with minimal impact.

Resources

For more information and to see examples and descriptions of design patterns for Microsoft Dynamics AX, go to: <http://msdn2.microsoft.com/en-us/library/ms940503.aspx>

How to Comply

Your documentation must include best practices and guidance on how to correctly customize and extend your application.

Test Methodology

During the in-lab test, the test vendor will verify that you have included customization procedures in Help text and in code comments.

Criteria for Passing

This requirement is mandatory. If the ISV does not document how to customize the application, the application will fail the test.

Upgrade and Maintenance

Your application must meet the following upgrade and maintenance requirements:

- [Requirement 10.1: The ISV must document all integration points.](#)
- [Requirement 10.2: The ISV must provide database upgrade scripts.](#)
- [Requirement 10.3: The ISV must use file versioning for DLLs and ActiveX controls.](#)

10.1 The ISV Must Document All Integration Points

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Almost every line-of-business application interoperates with components outside the core Microsoft Dynamics platform. This interoperability may be in the form of AIF, COM integration, .NET interop, Web Services, BizTalk, or even SQL Server-generated messages.

Resources

For more information, see the documentation example for AIF integration with standard Microsoft Dynamics AX, available on MSDN at: <http://msdn.microsoft.com/en-us/library/bb496530.aspx>.

How to Comply

You must ship integration point documentation as part of your submission as well as document it during your TWC documentation.

Test Methodology

The test vendor will review documentation to determine that the ISV has documented the integration points of the application.

Criteria for Passing

This requirement is mandatory. If the ISV does not document the integration points, the application will fail the test.

10.2 The ISV Must Provide Database Upgrade Scripts

Type	Test Method	Technology	Solution Category		
Required*	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

**Required if the ISV application has been updated.*

Summary and Intent

Upgrades are an important part of business software; the intent of this section is to make certain that upgrades are relatively easy for partners by providing scripts that support database upgrades.

When new versions, service packs, and hot fixes are released, Microsoft Dynamics AX uses upgrade scripts to handle data upgrades. Similarly, you must provide upgrade scripts for your application if you change the data model structure.

Resources

See the white paper on how to write upgrade scripts at:

https://mbs.microsoft.com/partnersource/documentation/whitepapers/msdyax40_writing_upgrade_scripts_whitepaper.htm?printpage=false

How to Comply

Before you submit your application for certification, install the application, identify any changes to the database model, and compare the upgrade scripts with the data model changes to make sure that your have accurate upgrade scripts for all changes to the application.

When you submit your application for certification, include the upgrade scripts.

If you rely on existing AX upgrade scripts and do not require upgrade scripts specific to your solution, it is sufficient to reference AX upgrade procedures for the module(s) your solution is integrated with.

Test Methodology

During the in-lab test, the test vendor will run your upgrade scripts and then test the application against the database to verify that the scripts function correctly.

Criteria for Passing

This requirement is mandatory. If the ISV does not provide database upgrade scripts, the application will fail the test.

10.3 The ISV Must Use File Versioning for DLLs and ActiveX Controls

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

All executable files, such as DLLs and ActiveX controls, must have versioning metadata associated with them. Installation programs use this metadata to confirm that correct versions are in place before applying an upgrade, service pack, or hot fix. Without this versioning information, an installation program could corrupt the system by applying changes that cannot synchronize with the file that is currently installed. Additionally, if a shared component fails, correct file version information lets a customer identify the associated application and file producer. The file's producer is the only entity that can regress the file; therefore, the metadata must also include the company name.

Resources

All DLLs and ActiveX controls must be file versioned. See [requirement 7.2](#) for more information on DLLs and ActiveX controls.

How to Comply

Before you submit your application for testing, examine the file information for each DLL and ActiveX control to verify that it includes the product name, company name, and file version number.

Test Methodology

Review submitted code to determine if files contain the metadata with the following elements:

- Product name
- Company Name
- File Version Number

Criteria for Passing

This requirement is mandatory. If the ISV does not use file versioning, the application will fail the test.

Sustainability

Your application must meet the following sustainability requirements:

- [Requirement 11.1: The ISV must remove non-functioning code from the code base.](#)

11.1 The ISV Must Remove Non-Functioning Code from Code Base

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Non-functioning code is confusing to support engineers who are attempting to resolve a problem and to developers who may work on your application in the future. In addition, there is always the risk, however remote, that code artifacts can be executed accidentally by another application or process.

Resources

See the development requirement section on how to use the Best Practice tool. This tool identifies non-functioning code.

How to Comply

It is a good practice to remove all non-functioning code from an application before the application is released. Your test plans and test scripts should check for non-functioning code.

Note: If you receive best practices errors and you cannot fix them, provide a written justification for allowing the errors, and include it as part of your submission package.

Test Methodology

During the in-lab review, the test vendor will run the best practice tool to determine if non-functioning code has been removed.

Criteria for Passing

This requirement is mandatory. If the application contains non-functioning code, it will fail the test.

Trustworthy Computing

Your application must/should comply with the following trustworthy computing requirements:

- [12.1: During the implementation phase, the ISV must establish and follow security development best practices.](#)
- [12.2: Before development begins, ISV staff should complete security and SDL training](#)

To comply with trustworthy computing requirements and recommendations, follow the guidelines in Microsoft Dynamics AX Help, particularly the "Development Best Practices" section.

12.1 The ISV Must Establish and Follow Secure Development Best Practices

Type	Test Method	Technology	Solution Category		
Required	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Your application development team must establish, communicate, and follow specific best practices for developing secure code. This approach will help them detect and remove security defects as early as possible in the development process. There are a number of resources, tools, and processes that you can use to accomplish this goal. The time and effort you take to apply best practices early in the product development cycle can prevent a more costly response to security defects later in the development cycle, or even more painful responses after product release.

Resources

For more information, see the following:

- [MSDN Library: Creating Named Shared Memory](#)
- [Microsoft E-Learning Security courses](#)

How to Comply

For .NET Framework–based applications, follow these guidelines:

- C# compiler: Use the latest version of the Microsoft Visual Studio® .NET development system. Use the specified compiler and linker flags for .NET Framework–based applications.
- csc.exe: Use version 8.0.50727.42.
- .NET Framework: Use version 2.0.50727.
- FxCop: Use the latest version of FxCop to locate and fix all security-related issues.
- Follow the guidelines in the SDL Shared section (that is, have no writable PE shared sections).

For X++ applications, mitigate the use of dangerous X++ APIs.

Establish the following best practices as you develop your application:

- Build tools: Use the currently required (or later) versions of compilers and compile options for your development platform.
- Code analysis tools: Use the currently required (or later) versions of code analysis tools for native (C and C++) or managed (C#) code.
- Previously released applications that were compiled with the **/GS** switch using Visual Studio version 2003 or later: You should research and resolve and **/GS** switch crashes. System crashes produced by binaries compiled with the **GS** option are marked in a specific way when uploaded through the Watson service. The data uploaded to Microsoft indicates where the buffer overflow occurred in the Microsoft code and also confirms that it happened in a customer use scenario. Because the **GS** compiler option mitigates some types of buffer overrun exploits, but not all of them, it is very important that you fix any reported buffer overruns in your code. For more information, see [Writing Secure Code, Second Edition](#) and [SDL Buffer Overflow Watson Crash Requirements](#), which outlines the process and requirements for servicing **GS** crashes.
- Banned APIs: New native code (C and C++) not use banned versions of string buffer handling functions. Based on analysis of previous MSRC cases, you should avoid the use of certain APIs to reduce vulnerability. See [Writing Secure Code, Second Edition](#) for more information.
- Executable pages: Avoid using executable pages. Unless your technology requires executable pages (for example, just-in-time [JIT] compilers), do not use them. This requirement affects how you call the **VirtualAlloc()** APIs and also prohibits compiling with **/NXCOMPAT:NO**. This requirement applies to Win32 and Win64 applications only; it does not apply to WinCE.)
- Writable shared PE sections: Your shipped binaries should not contain sections marked as **shared**. Shared binaries are a potential security threat. Use properly secured dynamically created shared memory objects instead.

Test Methodology

The test vendor will verify that your application meets the trustworthy computing requirement, as follows:

- Tools and processes: The vendor will perform a qualitative review to make sure that required processes and tools are in place and that processes were successfully executed. (The vendor will not audit the output of your tools and processes.)
- FxCop: The vendor will run FxCop and will require that you fix all reported issues.
- .NET Framework SDL Shared Section requirements (no writable PE shared sections): The vendor will run tools and require that you fix all reported issues.
- Dangerous X++ APIs: The vendor will perform an audit of a representative sample of your code to make sure that all trustworthy computing errors are fixed.

Criteria for Passing

This requirement is mandatory. If the ISV application does not establish and follow secure development best practices, it will fail the test.

12.2 Before Development Begins, ISV Staff Should Complete Security and SDL Training

Type	Test Method	Technology	Solution Category		
Recommended	In-lab review	All	Simple	Complex	Hosted
			✓	✓	✓

Summary and Intent

Microsoft has adopted a process called the *Trustworthy Computing Security Development Life Cycle (SDL)* to help make sure that software development follows security best practices. Security best practices require that developers are aware of secure coding practices including threats and countermeasures, and that they use the most recent versions of the Microsoft Visual Studio® development system, the .NET Framework, ADO.NET, and other tools and technologies. These tools provide the greatest compatibility with the platform infrastructure and provide the latest security advancements. For example, if your developers use Visual C++, they should use the **/GS** switch to compile their code and the **/RTC1** switch to debug their code.

In addition, SDL requires that developers should fix critical bugs that compromise security and perform security code reviews based on guidelines and checklists described in Appendix D of *Writing Secure Code* by Michael Howard and David C. LeBlanc. These reviews help ensure that the software design meets minimal trustworthy computing standards.

Resources

For more information, see the following:

- [Microsoft Dynamics AX - Writing Secure X++ Code](#) white paper
- [Microsoft Trustworthy Computing Web site](#)
- [Howard, Michael and David C. LeBlanc. *Writing Secure Code*. Second Edition. Redmond, WA: Microsoft Press, 2002](#)
- [Microsoft Press: *Writing Secure Code Companion Content*](#)

How to Comply

To satisfy the education requirement, all developers should complete the following:

- Read *Writing Secure Code*, Second Edition, by Michael Howard and David C. LeBlanc
- Read the *Microsoft Dynamics AX - Writing Secure X++ Code* white paper
- Avoid Potential Security Issues: <http://msdn2.microsoft.com/en-us/library/aa625308.aspx>
- Complete the following two [Microsoft E-Learning Security courses](#):
 - Clinic 2806: Microsoft Security Guidance Training for Developers
 - Clinic 2807: Microsoft Security Guidance Training for Developers II

To verify that your staff has satisfied this requirement, prepare a checklist or training documentation (such as a training overview, a Microsoft PowerPoint® presentation, class handouts, or syllabus), and include these items in your test submission package.

Test Methodology

The test vendor will review your training documentation to verify that your staff has completed the appropriate training. The vendor will use a score card similar to the following table.

	Company	Developer Name	<i>Date Completed</i>
<i>Writing Secure Code</i>			
Clinic 2806			
Clinic 2807			
Other			

Criteria for Passing

This is a recommendation only. Failure to comply with this recommendation will not cause the application to automatically fail the test.

Appendix A: User Assistance UI Requirements

Microsoft Dynamics AX supports the following user assistance formats:

- **Procedure topics.** Procedural help appears in a separate Help window when users explicitly request it; for example, by pressing **F1**. Procedural help provides information about how to complete a user task.
- **Form topics.** Form help appears in a separate Help window when a user explicitly requests it (for example, by pressing **F1**). Form help provides high-level information about the purpose of the form and links to common tasks that use the form, and describes the purpose of the tabs, buttons, and fields on the form.
- **Short help (Status bar messages and on-screen tooltips).** Status bar messages provide additional information about a specific control when the focus is in a field or when the user clicks a button. Status bar messages appear on the left side of the status bar, located in the lower-left corner of the form. Short help also appears as an on-screen tip (tooltip) when the user moves the pointer over ("mouses over") a button.

Procedure Help Requirements

Procedure topics in a Microsoft Dynamics AX application must comply with the following requirements.

- UI 1.1 - mandatory: Provide instructions for completing one task or similar, related tasks.
- UI 1.2 - mandatory: Take the user's point of view, and use second person singular (for example, "In the following procedure, you will...") or imperative (for example, "From the Start menu, click Control Panel.") voice.
- UI 1.3 - mandatory: Begin the title with an imperative verb, such as "Create" or "Add," that summarizes the task.

Form Help Requirements

Form topics in a Microsoft Dynamics AX application must comply with the following requirements.

- UI 1.4 - mandatory: Explain the overall purpose of the form, when and why the form should be used, and the purpose of the tabs, buttons, and fields on the form.
- UI 1.5 - mandatory: Provide a form topic for each form.
- UI 1.6 - mandatory: Use the name of the form in the topic title, such as "Sales order".
- UI 1.7 - mandatory: Divide form topics into short sections, such as: title, introduction, tasks, and navigation.
- UI 1.8 - mandatory: Provide links to related form help topics at the end of the topic under the heading "See Also".

Short Help Requirements

Short help in a Microsoft Dynamics AX application must comply with the following requirements.

- UI 1.9 - mandatory: Briefly describe the purpose of the control or what information the user should enter. Rewording a control label is not helpful.
- UI 1.10 - mandatory: For fields in which the user enters information or selects an item from a list, or buttons that only open another form or dialog box, use a noun phrase, such as "Name of the user account."
- UI 1.11 - mandatory: For check boxes or buttons that represent a choice or command, use a verb phrase, such as "Open the selected transaction."
- UI 1.12 - mandatory: Do not write short help in the form of a question.
- UI 1.13 - mandatory: Do not use ending punctuation, even for full sentences, unless the text consists of more than one sentence
- UI 1.14 - mandatory: Do not exceed 100 characters.

Appendix B: User Experience Guidelines

The following guidelines contain both recommended and mandatory requirements. An ISV must follow 'mandatory' requirements or provide descriptive documentation of the reason for non-compliance. It is suggested that an ISV follow the recommended guidelines as well.

Requirements for Application Windows

The windows in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 1.1 – recommended: Do not change the window size when your application opens a new window. For example: the window size must not change if a user changes from table view to tree view. (A user can use the standard Microsoft Windows operating system tools and conventions to resize windows.)
- UE 1.2 - recommended: Make sure that application windows fit into an SVGA display with 800 x 600 pixels.
- UE 1.3 - recommended: By default, do not require horizontal scrolling for application windows.
- UE1.4 - recommended: Do not use absolute coordinates to specify the size and location of a new window. When a window opens it must automatically size itself within the open application window, without exceeding preset size limits for windows.
- UE 1.5 – recommended: You should not specify window position in absolute coordinates.

Navigation Layer

Navigation Pane Requirements

The navigation pane is the primary navigation tool in the UI and contains links to the area page, primary and secondary list pages, and primary forms.

The navigation pane in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 2.1 - recommended: The lower part of the navigation pane shows the modules in the application, and the top part shows the contents of the selected module.
- UE 2.2 – recommended: A navigation pane should be created for each module in the application.
- UE 2.3 – recommended: All list titles should be plural (e.g. "Sales Orders" not "Sales Order").
- UE 2.4 – recommended: All list titles should be short and concise.
- UE 2.5 – recommended: Use sentence capitalization.
- UE 2.6 – recommended: Lists should be in the same order as they would be used in a process (e.g. "Sales Quotations" comes before "Sales Orders" that comes before "Invoices").
- UE 2.7 – recommended: If there is no obvious order to the way the pages are used, then the list should be sorted by importance, with the most important first.

- UE 2.8 – recommended: Forms do not have to be placed in the bottom of the navigation pane - they can be mixed with the other links.
- UE 2.9 – recommended: Only two levels should be created in the navigation pane tree.
- UE 2.10 – recommended: The same name should be used for the list in the navigation pane that you use for the list page title.

Area Page Requirements

The area page shows all the content of the module grouped into sections. Each module has one area page. The area page is automatically generated.

List Page Requirements

The list page shows the records in a grid. The list page also contains the navigation pane and the action pane. The action pane contains the actions that are specific to the list page.

A primary list page is shown in the first level of the navigation pane tree and shows the "full" list of records for an entity type. A secondary list page is shown in the second level of the Navigation pane tree. It has a filter applied to it that makes it more specific than the primary list page.

A list page in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 2.11 - recommended: There must only be one Primary list page per entity, but there can be multiple Secondary list pages for each entity.
- UE 2.12 - recommended: Create only one list page per entity (and not an individual list page for each persona).
- UE 2.13 - recommended: If the list page is a primary list page then no filter should be applied unless one is required to make the data meaningful or manageable.
- UE 2.14 - recommended: If the list page is a secondary list page, then a filter should be applied to the list.
- UE 2.15 – recommended: Primary list pages should be named the name of the entity.
- UE 2.16 – recommended: Secondary list pages should be named something that relates to the filter of the page.
- UE 2.17 – recommended: There should be no best practice warnings generated for the list page.

For more information, see the "Best Practices: List Page" topic in the Developing for Microsoft Dynamics AX Help file.

Task Forms Requirements

The task forms in a Microsoft Dynamics AX application must comply with the following requirements:

- UE 3.1 – mandatory: Use the following form types:
 - Parent/Child Form - a form with two tab controls presenting a business entity (record) in which there is parent and child data that are tied to each other intrinsically.

- The parent tab includes:
 - A grid containing a list of business entities (parent records)
 - The tab pages for the parent tab
 - The button group for the parent tab
- The child tab includes;
 - A grid containing a list of the lines (child data) for the business entity (parent record) selected on the header tab.
 - The tab pages for the child tab
 - The button group for the child tab
 - The splitter control allows the user to make the parent and child tabs taller or shorter (relative to each other)
- There are two variations of this type of form. Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.
- BOM Form – a form where the records must be identified by a version number in order to be useful. The lines for the selected parent record and child records open in a separate window.
 - The header tab includes:
 - A grid that displays a list of records that exist in various versions
 - The tab pages for the header tab
 - The button group for the header tab
 - The version tab includes;
 - A grid that displays a list of the versions of the record
 - The tab pages for the version tab
 - The button group for the version tab
 - Please reference the Microsoft Dynamics™ User Experience Guidelines for an example and additional guidance.
- Single Data Source Form - a form for presenting data that users will perceive as originating from a single data source with multiple records.
 - A tab page identifies a group of information organized logically from the user's point of view. Tab pages include:
 - An Overview tab, which gives the user a quick summary of the contents of the table
 - A General tab, which contains the most frequently used fields for the table
 - Any other optional tab pages included should represent a logical grouping of information
 - The grid is the area containing columns and rows that displays the records for each header tab.

- The button controls are used to provide the user with a way to trigger an action. The default buttons are:
 - Transactions
 - Setup
 - Inquiries
- There are two variations of this type of form. Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.
- Parameter Form – a form for setting up modules; for specifying default values and referencing number sequences.
 - A tab page identifies a group of information organized logically from the user's point of view. There is no Overview tab. Tab pages include:
 - A General tab, which contains the most frequently used fields for the table
 - Any other optional tab pages included should represent a logical grouping of information
 - Button controls are used to provide the user with a way to trigger an action. The General tab does not contain a button control area.
 - Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Dialog Boxes

Function Dialog Boxes Requirements

A Function dialog box is used when the program needs input from the user before starting a report or a job.

Function dialog boxes in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 4.1 - recommended: Do not use grids on a function dialog box.
- UE 4.2 - recommended: If a function dialog box starts a process, make sure that it contains an OK button and a Cancel button, and have these exact labels.
- UE 4.3 - recommended: Make sure that the OK and Cancel buttons are the right-most buttons in the lower right-hand corner of the dialog box.
- UE 4.4 - recommended: Standard buttons must have these exact labels and be in the following order: Select, Default, Options.
- UE 4.5 - recommended: If the user can change the value of a query, the query info should be on the right side of the dialog and read-only.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Wizard Requirements

Wizards in a Microsoft Dynamics AX application must/should comply with the following requirements.




- UE 5.1 - mandatory: In a wizard, include a welcome page, one or more interior pages, and a successful completion page.
- UE 5.2 - mandatory: On the welcome page, explain the purpose of the wizard. Back, Next, and Cancel buttons are supplied automatically.
- UE 5.3 - mandatory: Interior pages provide the user the choices they need in order to carry out the wizard's task. Back, Next, and Cancel buttons are supplied automatically.
- UE 5.4 - mandatory: Disable the Next button on interior pages until the user has entered the information that is required to continue.
- UE 5.5 - mandatory: The completion page tells the user what will happen when they click Finish. The Finish button automatically replaces the Next button on the successful completion page.
- UE 5.6 - mandatory: Make sure that users can start the wizard from a visible control, such as from a button on a form or from the main menu. A wizard must not open on a CTRL+N.
- UE 5.7 - recommended: Make sure that the wizard pages are easy to understand. Include a limited amount of text, and ask only for information that is required for a simple version of the task.
- UE 5.8 - recommended: Include default values or settings whenever possible.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Message Boxes

Infolog Message Box Requirements

Infolog message boxes inform the user of important information related to their work with the program after execution of their entire current task is complete.

Infolog message boxes in a Microsoft Dynamics AX application must/should comply with the following requirements:




- UE 6.1 – mandatory: Make a distinction between the different message categories:
 - Use error messages to inform the user of an error that has already occurred
 - Use warning messages to alert the user to a condition or situation that may be impending
 - Use info messages to notify users of the results of a command
- UE 6.2 – mandatory: Identify the different types of messages with the appropriate icon:
 - Always use the error icon to identify error messages: 
 - Always use the warning icon to identify warning messages: 
 - Always use the information icon to identify info messages: 

- UE 6.3 – recommended: If the Infolog contains more than one type of message, use the icon with the most severe type in the top right corner of the message box.
- UE 6.4 - recommended: Make sure that critical and warning messages are specific and constructive. They must describe a way of solving the user’s problem.
- UE 6.5 - recommended: Make sure that a typical user can understand the message. Write the message clearly, without jargon or confusing terminology.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Modal Message Box Requirements

Modal message boxes inform the user of important information immediately, due to a serious situation.

Modal message boxes in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 6.6 – mandatory: Make a distinction between the different message categories:
 - Use error messages to inform the user of an error that has already occurred
 - Use warning messages to alert the user to a condition or situation that may be impending
 - Use info messages to notify users of the results of a command
- UE 6.7 – mandatory: Identify the different types of messages with the appropriate icon:
 - Always use the error icon to identify error messages: 
 - Always use the warning icon to identify warning messages: 
 - Always use the information icon to identify info messages: 
- UE 6.8 - recommended: Make sure that critical and warning messages are specific and constructive. They must describe a way of solving the user’s problem.
- UE 6.9 - recommended: Make sure that a typical user can understand the message. Write the message clearly, without jargon or confusing terminology.
- UE 6.10 – mandatory: Info messages must have an OK button only.
- UE 6.11 – recommended: Yes and No buttons should be used in a modal message box if the message contains a clearly phrased question to the user.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Controls

List Page Filters Requirements

All filters are built into the controls, so there are no requirements to adhere to. This section is included here for completeness. There are five filter types on a list page:

- The Tier-1 (or Quick) filter is always visible and is located above the grid in the right side. The user types in a word and selects a column to filter on using the dropdown menu.
- The Grid filter allows the user to filter on each of the fields in the row by expanding a "filter line" on the top of the existing rows in the grid. It is expanded by clicking the highlighted icon above.
- Filter by Field appears in the context menu for the grid.
- Filter by Selection appears in the context menu for the grid.
- The Advanced Filter/Sort appears in the context menu for the grid.

Please reference the Microsoft Dynamics™ User Experience Guidelines for examples.

List Page List View Requirements

A list view control is a non-editable grid that is found on List Pages. It can also be found on a Wizard page.

A list view in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 7.1 – mandatory: Never use a list view control on an area page.
- UE 7.2 – recommended: Amounts should be right-aligned. Other data types should be left-aligned.
- UE 7.3 – mandatory: The entire row of a list view is a link to a task page. Do not make individual fields in the list into links.
- UE 7.4 – recommended: Use black font in the list view.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

List Page Menus Requirements

There are three menus that appear on all list pages. They are automatically generated, so there are no requirements to adhere to. This section is included here for completeness.

- The Microsoft Dynamics Menu is located near the top left corner. It contains general commands for the application and has the same content on all pages.
- The Layout Menu is located near the top right corner. It contains all the functions that helps the user personalize the layout of the page.
- The Help Menu is also located near the top right corner. It contains all the Help functions and has the same content on all pages.

Please reference the Microsoft Dynamics™ User Experience Guidelines for examples.

Form Controls Requirements

All items added to a form are controls, i.e., grids, trees, text boxes, list boxes, option buttons, command buttons, and images. Controls display data, perform actions, or provide decoration. For example, you can use a text box on a form to display data, a command button on a form to open another form, or a picture on a form to show a company logo.

Form control properties in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 7.5 – recommended: The “Auto” value should be used for a control’s property whenever possible to ensure a uniform user interface.
- UE 7.6 – mandatory: Form control names must be unique per form.
- UE 7.7 – recommended: The “AutoDeclaration” property should be set to “Yes” for controls that are referenced from X++ code on the form.
- UE 7.8 – recommended: The “AutoDataGroup” property should be set to “Yes” for a Group control that has the “DataGroup” property set.
- UE 7.9 – mandatory: The “HelpText” property must be set for form controls that have the property. If you don’t enter a value for the HelpText property, it will be inherited from the field or the extended data type that the control is based on, if possible. So, while this is a mandatory requirement, you don’t have to specifically enter a HelpText label if it can be inherited from somewhere else.
- UE 7.10 – mandatory: The “Label” property must be set for form controls that have the property. If you don’t enter a value for the Label property, it will be inherited from the field or the extended data type that the control is based on, if possible. So, while this is a mandatory requirement, you don’t have to specifically enter a label if it can be inherited from somewhere else.
- UE 7.11 – recommended: Form controls that cannot be changed by the user should have the following properties set:
 - “AllowEdit” = “No”
 - “Skip” = “Yes”
 - “Enabled” = “Yes”
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

For more information, see the “Form Controls” topic in the Developing for Microsoft Dynamics AX Help file.

Commands

Action Pane Requirements

The action pane contains all the actions related to a list page.

- Action pane tabs are used to organize the actions in the action pane.
- The actions in each tab are organized into groups and each group has a title. When the action pane resizes smaller, the groups will be collapsed one-by-one from the right.
- Action pane buttons display the actions and can be either push buttons or menu buttons.
- There are labels for the tab, the group, and the actions.
- All actions in the Action pane have an icon.
- The tabs and buttons in the Actions pane will automatically be assigned Key Tips based on the first character of their label.

An action pane in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 8.1 – recommended: The action pane should display in its entirety in 1024x768 resolution (no collapsed groups).
- UE 8.2 – recommended: Avoid duplicating an action on multiple tabs unless it makes sense in the context of the tab's activities.
- UE 8.3 – recommended: Push buttons have an icon and a label. Use big icons for frequently used buttons and small icons for those that are used less frequently.
- UE 8.4 – recommended: Menu buttons contain several actions accessed through the dropdown menu on the button. Use big icons for frequently used buttons and small icons for those that are used less frequently.
- UE 8.5 – recommended: Re-use icons from Microsoft Dynamics AX 2009. If your action is unique and you need to design a new icon, use an icon with a similar style.
- UE 8.6 – recommended: Frequently used actions should have high priority (big icons and position to the left of the action pane) so they will not be collapsed in an overflow menu when space is limited.
- UE 8.7 – mandatory: Tab titles, group titles, and action titles must be title case.
- UE 8.8 – recommended: Tab titles, group titles, and action titles should not contain abbreviations or acronyms.
- UE 8.9 – recommended: Tab titles should be activity-based and imperative.
- UE 8.10 – mandatory: Tab titles must be less than 20 characters long.
- UE 8.11 – recommended: Tab titles should not be the same as the list page.
- UE 8.12 – mandatory: Group titles that are activity-based must be imperative.
- UE 8.13 – recommended; Group titles should not be the same as the tab.

- UE 8.14 – recommended: Group titles should be one or two words and should not exceed 20 characters.
- UE 8.15 – mandatory: Action titles must not use these prefixes if it is possible to view, edit, and add on the form:
 - Open
 - View
 - Edit
 - Modify
 - Add
 - Exceptions:
 - Action titles can use the prefix “View” if the form only allows viewing.
 - Action titles can use the prefix “Add” if the action is a specific action where you are adding an item.
- UE 8.16 – recommended: Action titles for actions that point to another list, should use plural in the action title.
- UE 8.17 – recommended: Action titles be the same as other buttons/actions that link to the same menu item, unless the existing name is ambiguous or misleading,
- UE 8.18 – recommended: Action titles should not exceed 25 characters.
- UE 8.19 – recommended: An empty group should not be displayed.
- UE 8.20 – recommended: A group should contain up to eight actions.
- UE 8.21 – mandatory: “Cut”, “Copy” or “Select All” must not be included in the Action pane. These are supported by keyboard shortcuts.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Requirements for Enterprise Portal

EP Page Types

Role Center Requirements

The EP role center is the page that is displayed each time a user starts up and should be set up for each role at the company. A role center has four zones and a zone can contain any number of role center controls.

A role center in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 9.1 – recommended: The role center content should be useful at a glance.

- UE 9.2 – recommended: The role center should display all its content on a 1024 x 768 monitor without requiring a scrollbar.
- UE 9.3 – recommended: The most important role center control should be placed in the top zone or middle left zone.
- UE 9.4 – recommended: A role center should have four to six role center controls and no more than two chart controls.
- UE 9.5 – mandatory: A default role center page must be created and displayed if a role does not have a role center set up for it.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Report Page Requirements

The layout of an EP report page is determined by the SRS report. It is a full-page report and consists of a header and one or more graphs and/or a matrix.

A report page in a Microsoft Dynamics AX application must comply with the following requirements:

- UE 9.6 – mandatory: The report page must open in a new browser popup window.
- UE 9.7 – mandatory: The report page must not have a Quick Launch, FactBoxes, or FastTabs.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

List Page Requirements

The EP list page is the primary mechanism for displaying Microsoft Dynamics AX 2009 content. A list page consists of a Page Title, Toolbar, List View, List Filter, and optional FactBoxes.

A list page in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 9.8 – mandatory: A list page must contain a Page Title, Toolbar, List View, and List Filter.
- UE 9.9 – recommended: There should be one list page per entity.
- UE 9.10 – recommended: There should be no more than six FactBoxes on a list page.
- UE 9.11 – recommended: The same list page can exist in multiple Activity sites, but should never be placed directly under the Role Center or directly under a Module site.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Process Page Requirements

An EP process page supports a process or activity in a company which includes multiple tasks and takes place over a period of time. A process page allows users to collaborate and share documents. The process

page can either look the same to all users or look different to different users. A process page consists of Introduction Text, Steps, and FactBoxes.

A process page in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 9.12 – recommended: There should be a maximum of 10 steps on a process page.
- UE 9.13 – recommended: Steps should be open by default.
- UE 9.14 – recommended: There should be no more than six FactBoxes on a process page.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Task Page Requirements

An EP task page is used for most tasks in the system, grouping content in FastTabs that can be expanded or collapsed. A task page can be in one of three modes: View, Overview, or Edit. A task page consists of a Page Title, Toolbar, Form control & FastTabs, Command buttons, and optional FactBoxes.

A task page in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 9.15 – recommended: There should be no more than 10 FastTabs on a task page.
- UE 9.16 – recommended: There should be no more than six FactBoxes on a task page.
- UE 9.17 – recommended: There should be no more than six command buttons on a task page.
- UE 9.18 – mandatory: A task page in Edit mode must have a Cancel button in the right bottom corner.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Wizard Page Requirements

An EP wizard page is a type of task page that takes the user through a series of simple steps. It should be used for infrequent or complex tasks that can be split into simple sub-tasks and should be performed in a certain sequence. A wizard page consists of a Step Overview, Instructional User Assistance Text, Fields and/or controls for the active step, and Command buttons.

A wizard page in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 9.19 – mandatory: A wizard page must contain a Step Overview, Instructional User Assistance Text, Fields and/or controls for the active step, and Command buttons.
- UE 9.20 – mandatory: A wizard page must not contain FactBoxes or FactTabs.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Popup Window Requirements

An EP popup window is used to ask the user for clarifying information or when the user has to do something to complete an action. Popup windows are modal and require user interaction before any navigation can proceed in the browser window. A popup window consists of Instructional User Assistance Text, Fields and/or controls, and Command buttons.

A popup window in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 9.21 – mandatory: A popup window must contain Instructional User Assistance Text, Fields and/or controls, and Command button(s).
- UE 9.22 – recommended: A popup window should not contain more than six fields.
- UE 9.23 – mandatory: A popup window must not contain OK or Cancel buttons.
- UE 9.24 – mandatory: On a popup with non-editable content, a Close button must be the only button on the window.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Role Center Controls

Activities Requirements

The activities control shows up to six graphical cues, which provide the user with an overview of their workload. The control should be used for tasks a user deals with multiple times a day or tasks that are business-critical.

An Activities role center control in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 10.1 – recommended: The activities control should contain tasks that the user performs many times a day.
- UE 10.2 – recommended: When more than one activities control exists on the Role Center, the title of the control should describe the kind of activities contained in it.
- UE 10.3 – recommended: The activities control should not be wider than two columns of the role center.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Quick Links Requirements

The quick links control is a modifiable list of links providing quick access to web links, features, and reports that the user needs to do their job.

A Quick Links role center control in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 10.4 – recommended: The quick links control should include links to content that is important, but not so important it needs to be included in the role center.
- UE 10.5 – recommended: If more than one quick links control exists on the role center, the title of each control should be different.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Work List Requirements

The work list control informs the user of the status of activities by indicating that an action is required or an item needs attention.

A Work List role center control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 10.6 – recommended: The work list control should be used when timely reviews or approvals are required.
- UE 10.7 – mandatory: There must be only one work list control per role center.
- UE 10.8 – recommended: The work list control should be displayed at 100% width.
- UE 10.9 – mandatory: The work list title must be “Work List”.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

AX Report Requirements

The AX report control displays a report on the role center. This is a legacy control and has been replaced by the SRS Report control in Microsoft Dynamics AX 2009.

An AX Report role center control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 10.10 – mandatory: The SRS Report control must be used instead of this control for all new features.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

SRS Report Requirements

The SRS report control is a SharePoint web part that displays a SRS report in a role center.

A SRS Report role center control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 10.11 – mandatory: The SRS Report control must be used for all new features.

- UE 10.12 – recommended: The title of the SRS report control should match the name of the report.
- UE 10.13 – recommended: The report should remain centered in the web part regardless of the width of the role center zone where it is located.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Key Performance Indicator (KPI) Requirements

The KPI control is used for displaying information that the user needs to look at frequently and that can be summarized with an indicator, goal, value, and status.

A KPI role center control in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 10.14 – recommended: The title of the KPI role center control should be either "Business Overview" or a name that describes the content of the control.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Role Center Filter Requirements

The role center filter control allows advanced filtering of data in a list.

A Role Center Filter role center control in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 10.15 – recommended: The role center filter control should only be used on list pages.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Excel Web Access Requirements

The Excel web access control displays Excel content in the role center.

An Excel Web Access role center control in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 10.16 – recommended: The Excel web access control should not be wider than two columns of the role center.
- UE 10.17 – recommended: The title of the Excel web access control should describe the content of the control.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

EP Controls

Workflow Message Bar Requirements

The workflow message bar control is displayed when a workflow action is requested for the user, and can be used on: list pages, process pages, task pages, or wizard pages.

A Workflow Message Bar control in a Microsoft Dynamics AX application must comply with the following requirements:

- UE 11.1 – mandatory: The workflow message bar must be located directly below the toolbar.
- UE 11.2 – mandatory: If the page does not have a toolbar, the workflow message bar must be located directly below the page title.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Toolbar Requirements

The toolbar control contains buttons and/or menus relating to commands specific to the list page or task page. .

A Toolbar control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- Toolbar on a list page
 - New menu (optional)
 - UE 11.3 – mandatory: The first menu item must be “New <entity>”
 - UE 11.4 – recommended: All menu items should start with “New”
 - UE 11.5 – recommended: The menu should not contain more than five commands
 - UE 11.6 – mandatory: All menu items must open a task page in a popup window
 - Actions menu (optional)
 - UE 11.7 – mandatory: The first three menu items must be
 - View selected item
 - Edit selected item
 - Delete selected item
 - UE 11.8 – recommended: Workflow actions should be located below the mandatory items of the menu and divider
 - UE 11.9 – recommended: The menu should not contain more than eight actions
 - UE 11.10 – mandatory: All menu items must open a task page in a popup window

- Reports menu (optional)
 - UE 11.11 – recommended: The menu should contain reports related to the list type
 - UE 11.12 – recommended: The menu should not contain more than eight reports
 - UE 11.13 – mandatory: All menu items must open a report page in a popup window
- Related information menu (optional)
 - UE 11.14 – recommended: The menu should provide links to other relevant list pages which are filtered to the selected record
 - UE 11.15 – recommended: The menu should not contain more than eight items
 - UE 11.16 – mandatory: All menu items must open a list page in a the same window
- Toolbar on a task page in view mode
 - Edit command (mandatory)
 - UE 11.17 – mandatory: The first command must be “Edit <entity>” and change the mode to edit mode
 - New menu (optional)
 - See UE 11.3 – 11.5 above
 - UE 11.18 – mandatory: All menu items must open a task page in the same window
 - Actions menu (optional)
 - UE 11.19 – mandatory: The first two menu items must be “Edit <entity>” and “Delete <entity>”if the actions apply to the content of the task page
 - UE 11.20 – recommended: Workflow commands should not be located below the mandatory items of the menu and divider
 - UE 11.21 – recommended: The menu should not contain more than eight actions
 - UE 11.22 – mandatory: All menu items must open a task page in the same window
 - Related information menu (optional)
 - UE 11.23 – recommended: The menu should provide links to other places relevant to the task
 - UE 11.24 – recommended: The menu should not contain more than eight items
 - UE 11.25 – mandatory: All menu items must open a list page in a new popup window
- Toolbar on a task page in edit mode
 - New menu (optional)

- See UE 11.3 – 11.5 and 11.18 above
- Actions menu (optional)
 - See UE 11.19 – 11.22 above
- Related information menu (optional)
 - See UE 11.23 – 11.25 above
- Attach file button (optional)
 - UE 11.26 – mandatory: The attach file button must allow the user to select a file to attach
- UE 11.27 – recommended: The toolbar should not contain Submit or Workflow commands
- Toolbar on an entity overview page
 - Edit command (mandatory)
 - UE 11.28 – mandatory: The first command must be “Edit <entity>” and display the task page in edit mode in the same window
 - New menu (optional)
 - See UE 11.3 – 11.5 and 11.18 above
 - Actions menu (optional)
 - UE 11.29 – recommended: The menu should contain actions relevant to the entity overview
 - UE 11.30 – recommended: Workflow commands should not be located on this menu
 - UE 11.31 – recommended: The menu should not contain more than eight actions
 - UE 11.32 – mandatory: All menu items must open a task page in the same window
 - Related information menu (optional)
 - UE 11.33 – recommended: The menu should provide links to other places relevant to the entity
 - UE 11.34 – recommended: The menu should not contain more than eight items
 - UE 11.35 – mandatory: All menu items must open a list page in a new popup window
- UE 11.36 – mandatory: The toolbar must not be placed on role center pages, area pages, process pages, or wizard pages.
- UE 11.37 – mandatory: The toolbar must not be placed in role center or EP controls.
- UE 11.38 – mandatory: Toolbar content must remain the same regardless of what is selected on the page.

- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Infolog Requirements

The Infolog control displays important messages that the user must pay special attention to and is used on list pages, process pages, task pages, and wizard pages. The Infolog appears automatically when there is an information message, warning or error on the page.

An Infolog control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 11.39 – recommended: The infolog control should be placed directly below the page title.
- UE 11.40 – recommended: Message text should be limited to one sentence in length and should be clear and concise.
- UE 11.41 – mandatory: The infolog must not be used for workflow messages.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

List View Requirements

A list view control is a non-editable grid that is found on List Pages. It can also be found on process pages, task pages, and wizard pages.

A List View control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 11.42 – mandatory: A list view must not be used on area pages or role centers.
- See UE 7.2 – 7.4 above
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Editable List Requirements

An editable list control allows data to be edited directly in the list. It is used on process pages, task pages, and wizard pages.

An Editable List control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

List Filter Requirements

The list filter control allows advanced filtering of data on a list page.

- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Form & FastTab Requirements

A form control is a container that contains several sections called FastTabs. It is used on process pages, task pages, and wizard pages. FastTabs are used instead of tabs in the main content area of a task page.

FastTabs change between view and edit mode depending on the state of the task page. Each FastTab contains a title and a summary in the header area, although the content of FastTabs may vary considerably. Two common types of FastTabs content are: fields and basic data controls, and list control/editable list.

A Form & FastTab control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- Fields and basic data controls
 - UE 11.43 – recommended: Content should be grouped, each with a header.
 - UE 11.44 – recommended: Groups should be separated with an extra empty line to create visual separation.
 - UE 11.45 – recommended: There should be two columns, but there can be up to four columns.
 - UE 11.46 – recommended: All column labels should be left-aligned.
 - UE 11.47 – recommended: Text fields should be left-aligned.
 - UE 11.48 – recommended: Number fields should be right-aligned.
 - UE 11.49 – recommended: Input fields should be text fields, dropdown menus, or lookup controls.
 - UE 11.50 – recommended: Required fields should be marked with a red asterisk following the label.
- List control or editable list
 - UE 11.51 – recommended: Actions related to the list in the FastTab should be listed directly below the list.
- UE 11.52 – mandatory: One FastTab must not control the content of other FastTabs on the control.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Lookup Requirements

A lookup control presents data in a list with multiple columns and is used so the user can choose a value for a field. A lookup control features a search field and a pagination pane (if required). It is used on process pages, task pages, wizard pages, and popup windows.

A Lookup control in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 11.53 – recommended: A lookup should have maximum of five columns.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

FactBox Requirements

A FactBox control appears on the right side of the screen and displays additional information about the page. It is used on list pages, process pages, and task pages. FactBoxes can display information about the entire page or for a selected line on a list page. Standard FactBoxes display text fields or links (including enhanced links). FactBoxes can also be created using asp.net to display charts, SRS report controls, or 3rd party data.

A FactBox control in a Microsoft Dynamics AX application must/should comply with the following requirements:

- UE 11.54 – recommended: There should not be more than six FactBoxes on a page.
- UE 11.55 – recommended: A FactBox should not have more than eight lines of data.
- UE 11.56 – mandatory: A FactBox must not be used for displaying Help text.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Enhanced Preview Requirements

An enhanced preview control provides supplemental information or actions related to an item in a FactBox. The preview is opened when a user hovers over an enhanced preview link, which is designated with three green dots. There are two types of enhanced preview controls: fields and list. They both have a title, content area, and link area.

An Enhanced Preview control in a Microsoft Dynamics AX application must comply with the following requirements:

- UE 11.57 – mandatory: An enhanced preview must show what the user will see if they click the link.
- UE 11.58 – mandatory: An enhanced preview fields control must not display more than 10 fields.
- UE 11.59 – mandatory: An enhanced preview list control must not display more than five rows.
- UE 11.60 – mandatory: An enhanced preview list control must not have a horizontal scrollbar.
- UE 11.61 – mandatory: An enhanced preview list control must not be editable.

- UE 11.62 – mandatory: An enhanced preview must not have more than six links in the link area.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

EP Naming

Capitalization Requirements

There are two forms of capitalization in Enterprise Portal: title caps and sentence caps.

Title caps means using uppercase for the first letter of the first and last words in the title, as well as all words in-between – with the exception of articles, coordinating conjunctions, and prepositions.

Sentence caps means using uppercase for the first letter of the first word in the title, with other words in lowercase – with the exception of proper nouns.

EP Capitalization in a Microsoft Dynamics AX application should comply with the following requirements:

- UE 12.1 – recommended: Title caps should be used for: Breadcrumb navigation, Button names, Toolbar menus and menu items, Page titles, FastTab (section) names, Tab names in the top link bar, and Title bars (window title).
- UE 12.2 – recommended: Sentence caps should be used for: Check box labels, Combo box and Spin box labels, Descriptive text, Group box headings, List labels, List items, Radio button labels, Scrollbar labels, Slider labels, and Text box labels.
- UE 12.3 – recommended: Link labels should follow their target's capitalization.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Role Center Naming Convention Requirements

The Role Center Naming Conventions cover General Principles, Page Names, User Profile Names, Charts, Work Lists, KPIs, Cues, and Quick Links.

Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

EP Navigation

Top Link Bar Requirements

The top link bar consists of tabs at the top of the page for navigating through the portal.

A Top Link Bar in a Microsoft Dynamics AX application must comply with the following requirements:

- UE 13.1 – mandatory: The first tab must be labeled Home and opens the Role Center.
- UE 13.2 – mandatory: The other tab(s) must be labeled with the names of the Module sites (departments) that are relevant to the role of the user of the portal and open the area page of the module.

- UE 13.3 – mandatory: The icon located directly below the top link bar must reflect the tab currently selected by the user.
- UE 13.4 – mandatory: The top link bar must not hide or disable tabs when the user navigates around the portal.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Quick Launch Requirements

Quick Launch provides a method of navigating to key areas of the module

Quick Launch navigation in a Microsoft Dynamics AX application must/should comply with the following requirements;

- UE 13.5 – recommended: The role center quick launch should include: View All Site Content, Favorite pages, and modules.
- UE 13.6 – recommended: The quick launch for area pages, list pages, and process pages should include: View All Site Content, Main pages, Activity sections, and Reports.
- UE 13.7 – mandatory: The quick launch must never contain links to task pages.
- UE 13.8 – mandatory: All items in the quick launch must have unique names.
- UE 13.9 – recommended: When an item is active (displayed in the content area) it should appear in black text in the quick launch.
- UE 13.10 – mandatory: One item at a time must be active in the quick launch.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Requirements for Reports

SRS Report – Full Page Requirements

A SRS Report is suitable for reading on-screen or for printing and can be launched from a quick link, quick launch, or a link from the role center.

SRS Reports in a Microsoft Dynamics AX application must/should comply with the following requirements;

- UE 14.1 – mandatory: Report filters must be displayed at the top of the report.
- UE 14.2 – recommended: Each report filter should have a label.
- UE 14.3 – recommended: The most important content should be located to the top left of the report.
- UE 14.4 – recommended: The report should be sized to fill the page without using scrollbars.
- Please reference the Microsoft Dynamics™ User Experience Guidelines for examples and additional guidance.

Requirements for Icons and Symbols

Icons and symbols in a Microsoft Dynamics AX application should comply with the following requirements.

- UE 15.1 - recommended: A typical user should be able to understand all icons and symbols.
- UE 15.2 - recommended: The icon style should comply with Microsoft Office guidelines.
- UE 15.3 - recommended: A screen tip should be included that explains the purpose of the icon.
- UE 15.4 - recommended: Icons and symbols should use color and should not use black, gray, and white only.
- UE 15.5 - recommended: Icons should have a transparent background.
- UE 15.6 - recommended: Text should not be used in icons and symbols because such icons and symbols are difficult to localize.
- UE 15.7 - recommended: Blinking icons or symbols should not be used.

Appendix C: Microsoft Dynamics AX Consistency Verification Test

Company:

Application:

Version:

Prepared by:

Date Prepared:

- Identify the intended purpose of the product.*

Specify what fundamental service 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 Application is an application for managing data warehouse that enables the user to define the data warehouse strategy in a multi-facility and multi-user environment.

Describe the product purpose:

- Identify primary functions.*

Term	Definition	Notes
Primary function	Any function so important that, in the estimation of a typical user, its inoperability or impairment would make the product unfit for its purpose.	<p>A function is <i>primary</i> if you can associate it with the purpose of the product and it is essential to that purpose.</p> <p>Primary functions define the product. For example, the function of adding text to a document in Microsoft Word is so important that the product would be useless without it. Groups of functions, 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 of the functions on that toolbar should be operable for the product to pass the test.</p>

Examples of primary functions:

- Manage cross-docking operations.
- Manage stock environment.
- Manage IT interoperability with a fast carrier company.

List all primary functions:

- Identify contributing functions.*

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 grounds for test 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.*

1. List five to 10 functions or groups of functions (preferably primary functions) for focused instability testing.
2. 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.
- Design and record a consistency verification test.*

Prerequisites:

List any action that must be performed before the consistency verification test can be executed.

Examples:

- Install Microsoft SQL Server 2005.
- Use radio frequency identification (RFID) to identify hardware.

Required Information:

List any information that a user must know to perform the consistency verification test.

Examples:

- User must log on as User x.
- User must know the product serial number.
- User must know the account passwords.

Test Procedure:

Complete the following procedure to test each primary function of the application. You must describe each step that is required to test a primary function. You can combine similar functions, if appropriate.

Example:

- Manage cross-docking operations.
 1. Define the goods involved.
 2. Determine the delivery locations.
 3. Determine the transport company.
- Manage the stock environment.
 1. On the menu, select **File** and then select **Save**.
 2. Type the file name in the field, select the location for the file, and then click **OK**.
- Manage IT interoperability with a fast carrier company.
 1. Configure database access.
 2. Define user rights and circuit approval.