

Guide to Customizing AMS™ Suite: Intelligent Device Manager for PROFIBUS™ Devices

**AMS™ Suite: Intelligent Device Manager Toolkit for
HART Device Developers, Release 10.5**

**AMS™ Suite: Intelligent Device Manager, Version
10.5**

Emerson Process Management

**October 2009
U.S. Manual PN:**



Disclaimer

The contents of this publication are presented for informational purposes only, and while every effort has been made to ensure their accuracy, they are not to be construed as warranties or guarantees, express or implied, regarding the products or services described herein or their use or applicability. We reserve the right to modify or improve the designs or specifications of such products at any time without notice.

Copyright and Trademark Information

©Emerson Process Management, 2009. All Rights Reserved.

The Emerson logo is a trademark and service mark of Emerson Electric Co.

AMS™, PlantWeb®, FIELDVUE®, SNAP-ON™, DeltaV® are marks of one of the Emerson group of companies.

HART® is a registered trademark of the HART Communication Foundation of Austin, Texas, USA.

PROFIBUS® is a registered trademark of PROFIBUS & PROFINET International (PI).

FOUNDATION™ is a mark of the Fieldbus Foundation of Austin, Texas, USA.

AMS Device Manager is part of the PlantWeb field-based architecture from Emerson Process Management.

All other marks are property of their respective owners.

Contents

	Overview of AMS Device Manager-Related Toolkits	Overview-1
	What is AMS Device Manager?	Overview-1
	Using AMS Device Manager	Overview-2
	AMS Device Manager Functions	Overview-2
	The Development Tools	Overview-4
	AMS Device Manager Toolkit for PROFIBUS Device Developers	Overview-4
	AMS Device Manager Toolkit for HART Device Developers	Overview-4
	AMS Device Manager Toolkit for Foundation Fieldbus Device Developers	Overview-5
	AMS Device Manager Toolkit for SNAP-ON Developers	Overview-6
	Prerequisites for Using the AMS Device Manager Toolkits	Overview-6
	Required Knowledge Base	Overview-6
	Other Things a Developer Should Know	Overview-6
	Device Directory Path	Overview-6
	Where to Find More Information	Overview-7
	Technical Support for AMS Device Manager	Overview-7
	AMS Device Manager-Related Abbreviations and Acronyms	Overview-8
Section 1	Introduction	1-1
	What is in this Toolkit?	1-2
	Toolkit Documents	1-2
	Toolkit Files	1-3
	Online Help Development Guidelines	1-3
	Creating a Map File for Your Device	1-4
	Who Should Use This Toolkit?	1-4
	AMS Device Manager Architecture	1-5
	Purpose of this Document	1-6
	What You Should Know	1-6
	Organization	1-7
	Conventions	1-8
	Coding Examples	1-8
	Keys on the Keyboard	1-8
	Filenames and Directories	1-8
	Commands	1-8
	Variables	1-9
	Titles of Other Publications	1-9
	What's Next?	1-9
	Integrating a Field Device into AMS Device Manager	1-9
	Purchasing AMS Device Manager	1-9
Section 2	Getting Started	2-1
	Overview of Procedures	2-2

	Customizing AMS Device Manager for a Field Device: Overview of Steps. 2-2	
	File Types Needed for Customizing AMS Device Manager	2-3
	DEVICE.DDL (Required).....	2-3
	A_B_C_D.PBO (Required).....	2-3
	FMS.INI (Know about)	2-3
	DDINSTAL.INI (Required).....	2-3
	.MRG (Required).....	2-3
Section 3	Creating a Device Installation Kit	3-1
	Files and Directories You Will Use	3-2
	Applications You Will Use.....	3-2
	Assumptions.....	3-3
	Overview of Steps	3-4
	Troubleshooting.....	3-11
Section 4	Installing and Testing Your Device Files	4-1
	Hardware You Will Use.....	4-1
	Assumptions.....	4-2
	Overview of Device Installation and Test Steps.....	4-2
	Log Files for Debugging Your Device	4-6
	Clean Database.....	4-6
Section 5	Integrating the Device into AMS Device Manager.....	5-1
	Methods for Integrating into AMS Device Manager.....	5-1
	Independent Integration into AMS Device Manager.....	5-1
	Full Integration into AMS Device Manager	5-2
	The Integration Timeline.....	5-2
	The Next Step.....	5-2
Appendix A	AMS Device Manager Device Context-Sensitive Help-getting help beyond DD help stringsA-1	
	Introduction.....	A-1
	Providing Advanced Help	A-2
	Assumptions	A-2
	1) Determine which EDD parameters will need Advanced Help	A-3
	2) Create the device.ini file [ADVANCED HELP] section	A-3
	3) Create the map file	A-3
	Create the project file	A-4
	Create the topics	A-4
	Create the table of contents	A-5
	Create the index	A-5

Compile	A-5
Test.....	A-5
Add the help files to your device folder.....	A-6
Writing Guidelines	A-6
Conventions.....	A-6
Content.....	A-6
Writing Style and Usage	A-7
Example of MORE button.....	A-13
Providing a Product Manual	A-13
Add the PDF to your device files folder	A-14
Test.....	A-14

Figures

Figure 3-1. Export Window	3-9
Figure 5-1. Example of a More button	A-13

Overview of AMS Device Manager-Related Toolkits

What is AMS Device Manager?

AMS Device Manager is plant asset management software from Emerson Process Management, for managing field instruments and digital valve controllers. Its comprehensive set of analysis and reporting tools provides a single application for predictive diagnostics, documentation, calibration management, and device configuration. Using AMS Device Manager gives you better visibility into the devices in the plant, resulting in faster startup and increased availability through more cost-effective maintenance and improved device performance.

AMS Device Manager: An Integrated Solution for Asset Management

AMS Device Manager provides an integrated operating environment that leverages the full capabilities of intelligent FOUNDATION™ fieldbus, PROFIBUS, and HART devices, including embedded data trending, charting and graphical display capabilities provided by enhanced EDDL technology. For a description of the enhanced EDDL features in the current version of AMS Device Manager, refer to Technical Paper 925 included with this toolkit. AMS Device Manager is based on open communication standards, and is a core component of the PlantWeb digital plant architecture.

Using AMS Device Manager

AMS Device Manager is installed on a PC as either a Server Plus station or a Client SC station. A single station system consists of a Server Plus station with no Client SC stations connected to it. A distributed system is a Server Plus Station with one or more Client SC stations connected to it, where all stations can view and access devices connected to other stations. AMS Device Manager also supports installation on compatible DeltaV™ workstations, providing DeltaV users with a consolidated, seamless user interface to all of their online HART, FOUNDATION fieldbus, PROFIBUS, and conventional devices.

AMS Device Manager workstations communicate with online hard-wired and wireless field devices using one or more host system interface, including the DeltaV or Ovation® host system interface, the HART multiplexer system interface, the Rosemount® 3420 High-Speed Ethernet (HSE) interface, the Smart Wireless system interface, and many others. Multiple host system interfaces may be used with a single AMS Device Manager system.

AMS Device Manager Functions

AMS Device Manager lets you do the following:

- Display and modify device configurations. Compare configurations and transfer values from one configuration to another.
- Record and track events, using the optional Audit Trail. AMS Device Manager records events such as device configuration changes, username administration, status alert events, test/calibration activities, and database maintenance activities.
- Display and track alerts using Alert Monitor. Use Alert Monitor to notify you of problems with devices. Alert Monitor supports alerts for HART devices, fieldbus devices, PROFIBUS devices, and SNAP-ON applications. Each AMS Device Manager station can see the alerts for the entire distributed system.
- View device status conditions and diagnostics. With AMS Device Manager, you can view this information for devices connected to online systems such as DeltaV or Ovation. For fieldbus devices that support PlantWeb Alerts, you can view the alerts according to category and suppress or unsuppress individual alerts.

- View process variables for HART devices.
- Have a secure database for managing all of your plant instruments and digital valve controllers. You can construct a model of the devices in your plant using the Plant Locations hierarchy.
- Use electronic signatures to record which AMS Device Manager user made a configuration change.
- Perform loop tests and self-tests, and automatically document the test results.
- Streamline calibration procedures for as found and as left tests.
- Attach information such as drawings and service notes to individual devices using the Drawings/Notes feature.
- Track the conventional devices in your plant.
- Upload and download configurations to and from a Field Communicator or a 275 HART Communicator.
- Enhance AMS Device Manager functionality with value-added applications, such as the AMS ValveLink[®] SNAP-ON application.
- Restrict access to AMS Device Manager functions through the use of security permissions.
- Import and export information between AMS Device Manager systems, import data from other plant asset management systems, and export AMS Device Manager data to XML files.
- Access AMS Device Manager data through a standard Internet browser, with AMS Device Manager Web Client.
- Load AMS Device Manager data into business applications or systems such as the Emerson Process Management AMS Suite: Asset Portal[™] product, using Web Services.
- Expose live device data by means of the OPC interface.

The Development Tools

The AMS Device Manager toolkits provide the specific tools required by developers to integrate a field device into AMS Device Manager, and to provide additional device-specific functionality through AMS Device Manager. Currently, these toolkits include:

- The *AMS Suite: Intelligent Device Manager Toolkit for PROFIBUS Device Developers*
- The *AMS Suite: Intelligent Device Manager Toolkit for HART Device Developers*
- The *AMS Suite: Intelligent Device Manager Toolkit for FOUNDATION Fieldbus Device Developers*
- The *AMS Suite: Intelligent Device Manager Toolkit for SNAP-ON Developers*

AMS Device Manager Toolkit for PROFIBUS Device Developers

The development tools in the *AMS Suite: Intelligent Device Manager Toolkit for PROFIBUS Device Developers* include:

- Ancillary files, including sample code.
- AMS Device Manager standard and interoperable device descriptions.
- Instructions for editing a standard merge file for device information and creating a DD installation kit file.

Section 1 (“Introduction”) provides more information about what this toolkit contains and how to use it.

AMS Device Manager Toolkit for HART Device Developers

The development tools in the *AMS Suite: Intelligent Device Manager Toolkit for HART Device Developers* include:

- The *Guide to Customizing AMS Suite: Intelligent Device Manager for HART Devices* document that contains step-by-step procedures to ensure that a device will work with AMS Device Manager.
- The toolkit files to make your development tasks easier by providing resources such as precoded functions and libraries, thus alleviating many of the labor-intensive development tasks. These files include Device Windows Resource project files.

Note: *The AMS Suite: Intelligent Device Manager Toolkit for HART Developers, is available separately through your Emerson Process Management Sales/Service Office or*

on the Internet at <http://www2.emersonprocess.com/en-US/brands/amssuite/amsdevicemanager/Pages/DeveloperToolkits.aspx>.

**AMS Device Manager
Toolkit for FOUNDATION
Fieldbus Device
Developers**

The development tools in the *AMS Suite: Intelligent Device Manager Toolkit for FOUNDATION Fieldbus Device Developers* include:

- The *Guide to Customizing AMS Suite: Intelligent Device Manager for FOUNDATION Fieldbus Devices* document that contains step-by-step procedures to ensure that a device will work with AMS Device Manager.
- The toolkit files to make your development tasks easier by providing resources such as precoded functions and libraries, thus alleviating many of the labor-intensive development tasks. These files include Device Windows Resource project files.

Note: The AMS Suite: Intelligent Device Manager Toolkit for FOUNDATION Fieldbus Developers, is available separately through your Emerson Process Management Sales/Service Office or on the Internet at <http://www2.emersonprocess.com/en-US/brands/amssuite/amsdevicemanager/Pages/DeveloperToolkits.aspx>.

AMS Device Manager Toolkit for SNAP-ON Developers

Further functionality is achievable with devices integrated in AMS Device Manager through the use of SNAP-ON applications. These device-specific applications are layered on the core AMS Device Manager platform to perform additional functions for a connected device. The tools required to develop and test a SNAP-ON application are available in the *AMS Suite: Intelligent Device Manager Toolkit for SNAP-ON Developers*. These tools include:

- AMS Device Manager Add-in Utility that teaches AMS Device Manager to recognize new SNAP-ON applications.
- Ancillary files, including sample code and AMS Device Manager demonstration software, on CD-ROM.

Familiarity with the AMS Device Manager OPC Server architecture is required for writing SNAP-ON applications.

Note: *The AMS Suite: Intelligent Device Manager Toolkit for SNAP-ON Developers, is available separately through your Emerson Process Management Sales/Service Office.*

Prerequisites for Using the AMS Device Manager Toolkits

To be able to effectively use the AMS Device Manager toolkits to integrate HART devices, developers must have the knowledge described in the following paragraphs.

Required Knowledge Base

Developers must understand how to create a Device Description (DD) in the Device Description Language (DDL).

Other Things a Developer Should Know

For basic information on installing, starting up, and using AMS Device Manager, refer to the *AMS Suite: Intelligent Device Manager Installation Guide* and Books Online. (Following AMS Device Manager installation, these are available by selecting **Start**→**Programs**→**AMS Device Manager**→**Help** from the Windows taskbar.)

Device Directory Path

Devices are installed in the C:\AMS\DEVICES directory.

The Device Installation Kit will install the device in the appropriate subdirectory. In this document, the path to the DEVICES directory is represented as "...DEVICES".

Where to Find More Information

The following table summarizes the reference material available for AMS Device Manager developers.

Reference	Part Number	Source	Address	Phone/Fax	Web Address
<i>AMS Device Manager Toolkit for PROFIBUS® Device Developers</i>		Emerson Process Management	12001 Technology Drive Eden Prairie, MN 55344	P:1-800-833-8314	AssetWeb.com
<i>AMS Suite: Intelligent Device Manager Installation Guide Software Version 10.0</i>	10P58249001				AssetWeb.com
<i>Microsoft Windows User Experience: Official Guidelines for User Interface Developers and Designers</i>		Windows SDK (part of the MSDN Library)			msdn.microsoft.com
<i>PNO</i>		PROFIBUS & PROFINET International			

Technical Support for AMS Device Manager

For technical assistance in using the AMS Device Manager Toolkits or the AMS Device Manager sample code contained on the Toolkit CD-ROM, or to have a HART DD tokenized for use with AMS Device Manager, contact your Emerson representative, call 1-800-833-8314, or email amsdevicemanagertoolkit@emerson.com.

AMS Device Manager-Related Abbreviations and Acronyms

API	Applications Program Interface
CMA	Configuration Management Application
CSV	Comma Separated Value
DBCS	Double-Byte Character Set
DBMS	Database Management System
DCI	Device Communications Interface
DCS	Distributed Control System
DD	Device Description
DDL	Device Description Language
DDOD	Device Description Object Dictionary
EDDL	Electronic Device Description Language
GMT	Greenwich Mean Time
ISO	International Standardization Organization
LRV	Lower Range Value
MBCS	Multibyte Character Set
ODBC	Open Database Connectivity
OLE	Object Linking and Embedding
PC	Personal Computer
UCT	Universal Coordinated Time
URV	Upper Range Value

Note: *A glossary of terms used in AMS Device Manager can be found in Books Online.*

Introduction

This section introduces the *AMS Toolkit for PROFIBUS® Device Developers* and this guide. The section:

- Explains what is in the toolkit and who should use it.
- Briefly describes the AMS Device Manager system components and how you can extend the AMS Device Manager functionality.
- Describes the purpose of this document, how it is organized, and the conventions used in the document.
- Directs you to the next steps in customizing the AMS Device Manager application.

The parts of the toolkit that you will use to develop your devices is determined by the version of AMS Device Manager you are targeting.

Device Screens
(Required)

These are defined by an Enhanced EDDL DD with PC-based menus defined. The root menus are `process_variables_root_menu`, `device_root_menu`, and `diagnostic_root_menu`.

Merge File
(Required)

The minimum information in this file defines the name of your company and device to the system. You can also provide a template for your device.

- DDINSTALL.INI File (Required) This file provides the installation instructions for your device.

- A_B_C_D.INI File (Optional) This file is necessary for enhanced EDDL devices.

- Help File (Optional) Provides context sensitive help for your device, beyond DD help.

- PDF Manual (Optional) A PDF user manual can be provided for your device. These files will be launched from the Help right-click context menu.

Information about specific tools to use for the various AMS Device Manager versions is contained in Table 1-1 on page 1-2.

Table 1-1. AMS Device Manager Version Requirements

Windows Resource File requirements		Enhanced EDDL device only								
Devices		Enhanced EDDL Device, config, status and PV screens	Enhanced EDDL Device, compare configuration screen	Icon Source	Pre and Post Commit methods definition	Critical Parameters	Offline wizard	Enhanced Help parameter control	Device Menu	Device Method list
AMS Device Manager Version	Supports Enhanced EDDL	DDL Root_Menus	DDL Device_Root_Menu	DDL	DDL	DDL	DDL	A_B_C_D.INI file	DDL Root_Menus	DDL Root_Menus
11.0	Yes									

What is in this Toolkit?

This toolkit includes documents and associated electronic files, which together provide the software development and support services tools you need to integrate a PROFIBUS device into the AMS Device Manager application.

Toolkit Documents

The toolkit documents include:

- This guide, which provides step-by-step procedures for field device developers to follow in preparation for integrating a field device into the AMS Device Manager application. It gives instructions for creating a device installation kit and verifying the device interface.

- TP-925—Details an Enhanced EDDL functionality supported in each release.

Toolkit Files

The toolkit electronic files on the Internet (http://www.emersonprocess.com/optimize/technology/tectkt_include.htm) provide the following:

- Sample DD
- Sample files
- Template files
- Reference documents

Information about these toolkit files and how to use them is provided in the remainder of this document.

Online Help Development Guidelines

You should provide online Help for devices integrated into the AMS Device Manager application. Refer to **Appendix A: “AMS Device Manager Device Context-Sensitive Help-getting help beyond DD help strings”** for a description of the steps involved.

This document contains specifications for creating online Help systems for AMS Device Manager device applications. It is designed to assist experienced Help developers who are responsible for creating Help systems that run in the AMS Device Manager environment.

Note: *These guidelines are only for AMS Device Manager Help systems; they are not intended to be a detailed guide to the Help development process.*

Creating a Map File for Your Device

One of the optional features of AMS Device Manager is the Calibration Assistant SNAP-ON Application. In order to implement the uploading and downloading of information to/from intelligent calibrators, it is necessary to present the information in a common format. The majority of the information can be found in the HART Device Description for a device. Because the HART specification does not require that all parameters be in a common format, AMS Device Manager has implemented functionality that reads a .MAP file created for each device and puts the device parameters into the common format needed to manage calibration operations in your plant.

For your device to operate successfully with the AMS Device Manager Calibration Assistant SNAP-ON Application, it is necessary for you to develop a map file. To develop this file, please refer to the *MAP File Specification* and MAPManager utility included with this toolkit.

If you are developing a new Device Description to be integrated into AMS Device Manager, you must develop, execute, and approve a test to determine if the quality of the Device Description is acceptable for release to production. Similarly, if you subsequently modify your Device Description, you must repeat the testing and provide approval for release to AMS Device Manager production.

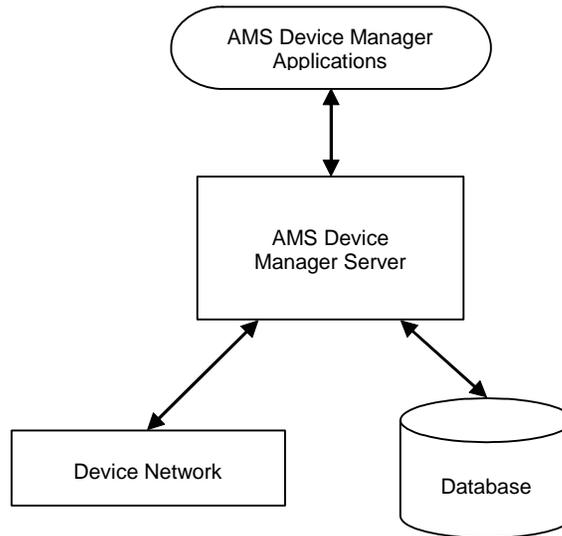
Who Should Use This Toolkit?

The *AMS Toolkit for PROFIBUS® Device Developers* was developed for and is intended specifically for use by the PROFIBUS device developer who wants to integrate a PROFIBUS device into the AMS Device Manager functionality. The level of the information provided in this toolkit assumes that the user understands how to write device descriptions (DDs) in the Device Description Language (DDL).

This toolkit does not provide instruction in programming, creating DDs, or tokenizing. If you need reference material on those subjects, refer to the Tokenizer Readme.txt.

AMS Device Manager Architecture

The following diagram is a high-level organizational view of the AMS Device Manager system components.



AMS Device Manager applications communicate with the servers, which in turn communicate with the device network and the database. Users conduct all device management tasks through the various AMS Device Manager applications.

Access to the device network and the database information is managed by the AMS Device Manager Server (plant server and file server). Because AMS Device Manager uses a client-server architecture, applications do not have to physically reside on the same computer as the servers. This allows you to run AMS Device Manager applications (clients) on machines remote from the servers over a TCP/IP network.

The device network in the diagram represents the actual field devices.

The database is the repository for historical device data and device configuration information.

You can integrate a smart field device into the AMS Device Manager application by modifying its DD. This document explains how to do this and provides sample files to assist you.

Purpose of this Document

This document provides a detailed, step-by-step guide for PROFIBUS device developers to follow in customizing AMS Device Manager in order to integrate a field device into the AMS Device Manager interface. It does not explain the basic functions of AMS Device Manager or the details behind the AMS Device Manager interfaces. For more information about basic AMS Device Manager functionality, refer to the *AMS Suite: Intelligent Device Manager Installation Guide* or Books Online.

What You Should Know

This document assumes that you understand how to use the features and functions of Windows operating systems. If you need more information about these operating systems, refer to the documentation that came with the systems.

This document also assumes that you have knowledge about the PROFIBUS Tokenizer and Development Environment.

Organization

This remainder of this document consists of the following sections:

- Section 2: “Getting Started” presents an overview of the procedures to follow when customizing AMS Device Manager for a device, and introduces the supporting files used in the procedures.
- Section 3: “Creating a Device Installation Kit” explains how to modify files and implement an installation program that allows your device to be installed into AMS Device Manager.
- Section 4: “Installing and Testing Your Device Files” explains how to install your device into an existing AMS Device Manager system and provides information on how to test it using AMS Device Manager.
- Section 5: “Integrating the Device into AMS Device Manager” explains how to integrate your device into AMS Device Manager once the installation kit has been created.
- Appendix A: “AMS Device Manager Device Context Sensitive Help-getting help beyond DD help strings” describes the procedure for linking device Help to context sensitive fields in the resource file.

Conventions

This document uses the typographic conventions described below.

Coding Examples

Examples of code, provided throughout the document, are shown in *Courier* (“typewriter”) monospaced font, with a rule line above and below the code. Code examples are aligned to the left margin of the page to provide space for longer lines of code. The following is an example:

```
// Define Style bits used for control types.
#define PS_EDIT          1L      // Edit control
#define PS_COMBO        2L      // Combo box
#define PS_LIST         4L      // List box
#define PS_GROUP        8L      // Group box with radio buttons.
#define PS_GAUGE       8192L    // Gauge Graphical Display
```

Keys on the Keyboard

Keyboard keys are shown in **SMALL CAPITAL** letters:

ENTER

Key combinations are shown using the plus (+) sign:

SHIFT + F1 indicates that you should press and hold down the SHIFT key, press F1, then release both keys.

Key sequences are shown using a comma (,):

ESC, F1 indicates you should press ESC, release it, then press F1.

Filenames and Directories

All files and filenames in this document use MS-DOS or Windows naming conventions. Filenames and directories are represented in **ALL CAPITAL** letters:

C:\WINDOWS\FMS.INI refers to the file FMS.INI, located in the WINDOWS directory on the C drive.

Commands

Commands you must enter are shown in **bold Courier** type, within the normal text margins:

DIR C:\AMSBLD\RELEASE*.* means you should type the command shown in the example, letter for letter.

The phrase “Enter the following command:” means that you should type the command shown, then press the RETURN or ENTER key on the keyboard.

Variables

Variables in commands are shown in *italicized Courier* type:

COPY source destination means you should enter the command “copy,” but replace source and destination with your specific information.

Where necessary, the variables are defined in the text.

Variables other than in commands are shown in *italicized* text:

DEVICE.SYM refers to a file with the extension “.SYM,” but you should replace “*DEVICE*” with the actual name of your file.

Titles of Other Publications

Titles of other publications appear in italicized text, with the publication number in parenthesis beside the title:

Device Description Language Specification (HCF_SPEC-500).

What's Next?

Section 2: “Getting Started” will guide you through the development process step-by-step.

Integrating a Field Device into AMS Device Manager

To successfully integrate your device into the AMS Device Manager application, you must thoroughly understand your device DD. If you need to install the AMS Device Manager software, refer to the *AMS Suite: Intelligent Device Manager Installation Guide*. To verify your device interface with AMS Device Manager, refer to Section 4: “Installing and Testing Your Device Files”.

Purchasing AMS Device Manager

You must purchase an AMS Device Manager application package so that you can effectively test the device descriptions you create. To purchase a copy of the application, contact your Emerson Process Management Sales/Service Office.

Getting Started

This section provides an overview of the customization procedures and introduces the files you will be working with.

Overview of Procedures

This guide explains how to customize AMS Device Manager for a PROFIBUS device.

Customizing AMS Device Manager for a Field Device: Overview of Steps

The following general steps are necessary to customize AMS Device Manager for a particular field device:

1. Modify relevant supporting files. The relevant supporting files are included on the Toolkit.
2. Create a “device installation kit” for installing your device into an existing AMS Device Manager system (see Section 3: “Creating a Device Installation Kit”). You must also test the installation kit itself.
3. Add a device template to the .MRG file.
4. Optional: create and integrate AMS Device Manager-compatible online Help files for your device. For further information on how to create and integrate these files, see Appendix A.

File Types Needed for Customizing AMS Device Manager

In the course of customizing AMS Device Manager for a device, you may need the types of files listed below. You will need to create some, modify some, and simply know about others, as noted in their descriptions. If you are developing device files using enhanced EDDL, some of the files are optional.

The Toolkit includes actual files or samples of the files you need to work with in customizing AMS Device Manager for a PROFIBUS device.

Note: *DEVICE, as part of a file name, refers to the base name of the device (usually the same as the base name of the DDL file).*

DEVICE.DDL
(Required)

The DD file, written in Device Description Language (DDL), contains information that describes your device to AMS Device Manager, including the device parameters and device methods.

A_B_C_D.PBO
(Required)

These files are generated as output by the DD tokenizer (A is the Manufacturer, B is the Device type, C is the Device Rev, and D is the DD Rev). You will need these files when you install your device into AMS Device Manager.

FMS.INI
(Know about)

The FMS.INI file is the AMS Device Manager initialization file. Your installation kit modifies it to indicate the presence of your tokenized device file, A_B_C_D.PBO when you start the AMS Device Manager application to test your device. Communications logging is enabled in this file.

DDINSTAL.INI
(Required)

The DDINSTAL.INI is the initialization file for the device installation portion of AMS Device Manager. You must modify this file to reflect your device.

.MRG
(Required)

The .MRG file is the database merge file you create that contains your device template. (You create a device template as part of the process of building a device installation kit.)

Creating a Device Installation Kit

This section explains how to create an installation kit that allows you to install your device into an existing AMS Device Manager system. You also need to create this installation kit to test your device interface.

Files and Directories You Will Use

The following files and directory structures are used to add your device to AMS Device Manager. Some are required, others are optional, and some you simply need to know about.

DEVICE.DDL
(Required)

These files are used as input to the tokenizer to generate the *DEVICE.PBO* and *DEVICE.SYM* files.

DEVICE.PBO and
DEVICE.SYM
(Required)

These files are generated as output by the DD tokenizer.

FMS.INI
(Know about)

The FMS.INI file is the AMS Device Manager initialization file. Your installation kit modifies it to indicate the presence of your tokenized device file, *A_B_C_D.PBO* when you start the AMS Device Manager application to test your device.

DDINSTAL.INI
(Required)

DDINSTAL.INI is the initialization file for the device installation portion of AMS Device Manager. You must modify this file to reflect your device.

DEVICE.MRG
(Required)

DEVICE.MRG is the database merge file you will create that contains your device template. (You create a device template as part of the process of building a device installation kit.) When you create the merge file, you will need to change the default name of the file, as explained later in this section.

...\DB\
(Know about)

This is the directory in which the AMS Device Manager database is located by default.

Applications You Will Use

You will use the following applications to add your device to AMS Device Manager:

- PROFIBUS Tokenizer – To generate the binary output file for your *DEVICE.DDL* file for adding the device to AMS Device Manager.

Note: *AMS Device Manager 10.5 requires files to be tokenized with a tokenizer version 1.0.0.10 or higher.*

- AMS Device Manager Add Device Type utility – To test your installation kit files.

- AMS Device Manager Network Configuration utility – To configure the hardware you use to connect your device to AMS Device Manager.
- AMS Device Manager – To create a device template and export that template to a merge file.

Assumptions

This section assumes that:

- You have installed AMS Device Manager.
If you need more information about installing AMS Device Manager or selecting a data source, see the *AMS Suite: Intelligent Device Manager Installation Guide*.
- You have a tokenized DD file.

7. Open the renamed TEMPLATE.MRG file in Notepad or Wordpad and modify the variables with text appropriate for your device. The TEMPLATE.MRG file looks like this:

<pre>// Export file, created 2003/12/03 15:13:51, J:\Workshop.ams_merge export { // Header section: MrgFileVersion 1.6, Time 2008/05/08 13:19:20, Description "Data from database FMS", Default Source "FMS export operation", // Definition section: // Manufacturers: manufacturer { Name "__MFG_NAME__", Protocol { Name "PROFIBUS-DP", ManIdentifier "__MFG_ID__" } } // Device Types: devicetype Y00001 { DeviceType "__DEVTYPE_ID__", Name "__DEVTYPE_NAME__", Description "__DEVTYPE_NAME__", Manufacturer "__MFG_NAME__", Protocol "PROFIBUS-DP" ManIdentifier } // Device Revisions: devicerev R00001 { DeviceTypeAlias Y00001, Name "__DEV_REV__", DeviceRevision "__DEV_REV__", MajorCategory "Unknown", MinorCategory "Unknown" ExtProperty { Name "GsdId" Value "_GSD_IDENT" } } // Device templates: template T00001 { Name "PROFIBUS-DP-Template", Type "T", ProtocolRevision 0, DeviceRevAlias R00001 } }</pre>	<p>Example variable entries:</p> <p>Control Techniques</p> <p>138</p> <p>2</p> <p>Commander SK</p> <p>Commander SK</p> <p>Control Techniques</p>
---	--

At this point, you have created a basic Device Installation Kit. The steps that follow allow you to take advantage of advanced features

in AMS Device Manager 10.5. The steps will vary depending on the version and features in the DD.

1. Copy the A_B_C_D.INI template file to the Device DD directory.
2. Rename the A_B_C_D.INI file the same as your .PBO file, but with the .INI extension.
3. If you have a .CHM file, add it to the Device DD directory. Update the DDINSTALL.INI and retest the Device Installation Kit in AMS Device Manager.
4. If you have a .PDF file for your device, add it to the Device DD directory. Update the DDINSTALL.INI and retest the Device Installation Kit in AMS Device Manager.
5. When all testing has been completed satisfactorily, go to page 3-8 to create a final merge file, with template.
6. Submit your tested and verified Device Installation Kit to the AMS Device Manager Development Team.

Updating the DDINSTALL.INI File

1. Under the [General] heading, there is a line for NumOfMfg. This is the number of manufacturers on the diskette. If you are the only manufacturer who will be using this Device Installation Kit, the line should be:

```
NumOfMfg=1
```

2. There is a heading for each Manufacturer, called [Mfg*n*] where *n* is the number of the manufacturer. If you are the only manufacturer using this install kit, the heading should be:

```
[Mfg1]
```

There are three lines in the [Mfg1] section; Name, ID and NumOfDev. Name represents the manufacturer name (for example, Fisher Controls); ID represents the 6-digit hexadecimal manufacturer ID (for example, 000013); NumOfDev represents the number of devices for this manufacturer on this Device Installation Kit (normally, this will only be one).

Using these examples, your [Mfg1] section would look like this:

```
[Mfg1]  
Name=Control Techniques  
ID=00008A  
NumOfDev=1
```

The entry beside “Name” will be manufacturer during add device type.

The entry beside “ID” is your hexadecimal manufacturer ID as assigned.

3. The heading after [Mfg1] represents the first device on the Device Installation Kit by that manufacturer. The heading is represented by the 6-digit hex ID of the manufacturer, a dot, and a 1 (for the first device). For example, the heading might be **[00008A.1]**.

Make sure this heading represents your manufacturer ID.

4. The first line under the ID heading is DevType. This is the 4-digit hex ID for the device type. For example, if the device type ID is **0002** in hex, the line would be:

```
DevType=0002
```

5. The second line is the description of the device type. For example, for a **Commander SK**, the line would be:

```
DevTypeDesc=Commander SK
```

6. The third line is a text description of the device revision. For example, for a device revision of **1**, this line would be:

```
DevTypeRevDesc=Rev 1
```

7. The fourth line is the actual revision of the device (2 digits). For example, for a device revision of **1**, this line would be:

```
DevTypeRev=01
```

8. The fifth line is a text description of the DD revision of the device. For a DD revision of **1**, this line would be:

```
DDRevDesc=DD Rev 1
```

9. The sixth line is the name of the PBO file for the installation kit.

```
PBO=8A_2_1_1.PBO
```

10. The ninth line is:

```
OtherDll=8A_2_1_1.INI (Optional)
```

This can also be used for the Device.BMP file.

11. The next few lines are the names and disk numbers of any general device help file, block help file, or other type of device help file (including pdf documentation). If you do not have any help files, use 'xxxx.chm' or 'xxxx.pdf'. Otherwise use your file names. For a device with no additional help files an example of these lines would be:

```
DeviceHelp=xxxx.pdf  
BlockHelp=xxxx.chm
```

12. The next line is the name and diskette location of the merge file. For example, if the merge file is named **3051C.MRG**, the line would be:

```
Template=8A_2_1_1.MRG
```

13. The final line indicates which AMS Device Manager versions your device operates with. For example, if your device is supported by AMS Device Manager versions 10.5-11.0, the line would be:

```
AmsVersionSupport=10.5, 11.0
```

14. Save your changes and close the DDINSTAL.INI file.
15. Copy the DDINSTAL.INI file to the folder in the device name directory (i.e., A:*devicename*\DDINSTAL.INI).

Create a Merge File and Edit it for Your Device

Once you have installed and scanned your device, you can create the template for your device. You must export the template into a merge file that will be part of the installation kit.

1. Start AMS Device Manager, if it is not already running.
2. Right-click on the Plant Database icon in either the Device Connection view or in the AMS Device Manager Explorer view.
3. From the Plant Database context menu, select **Export** → **To AMS Device Manager Export File**. The screen shows the Export window (Figure 3-1).

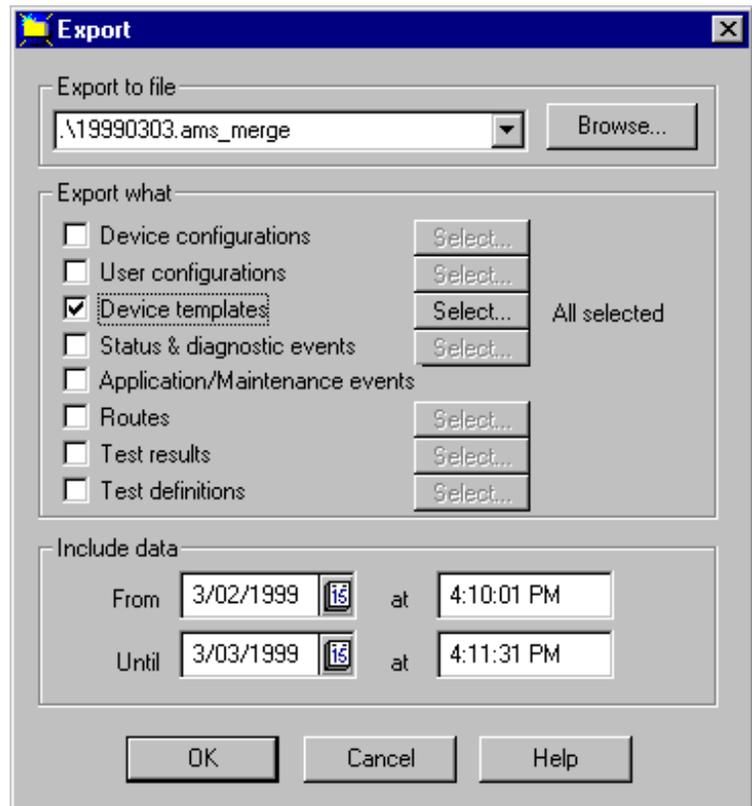


Figure 3-1. Export Window

4. Under Export What, deselect every option except Device templates (by default, all export options are checked).
5. Click **Select...** opposite “Device templates”. The screen shows which device templates are selected for export.
6. Make sure only your new device template is selected.
7. Click **OK** to close the Selection window. The screen returns to the Export window.
8. In the box below Export to file, enter the name and path of the file to which you want your template exported. Select a path you can easily find and change the name to the format *DEVICE.MRG*.
9. In the “Include data” section of the Export window, set the From and Until dates to cover the time period from when you created the database to the current time. Normally, setting the From date back one day and the Until date forward one day is sufficient.

Note: If you do not include a large enough time-span, your merge file will not contain all the necessary information.

10. Click **OK** to begin the export.
11. Once the export has completed, close AMS Device Manager.
12. Open the merge file you created (using WordPad or another text editor).
13. Search the file for “Event Section.”
14. Locate the User line in the Event Section.
15. If User is not set to “Install”, change it to “Install”.
16. Close the merge file and save the changes. Save the file as an ASCII file with line breaks.

Troubleshooting

If you have any trouble with your Device Installation Kit, the following troubleshooting suggestions may be helpful:

- If the device for which the new template is being created is displayed as a generic device by AMS Device Manager, check if the FMS.INI file has been properly updated.
- If the merge file contains more than one manufacturer, device revision, or device template, this may be caused by the selection of more templates than required as items to be imported.
- If AMS Device Manager does not show the live device on the interface, do the following:
 - Rebuild Hierarchy.
 - Scan New Devices.
- If AMS Device Manager issues error messages when it tries to populate the database with the new device or when making the template for the new device, do the following:
 - Run the AMS Device Manager Database Verify/Repair operation (select **Start**→**Programs**→**AMS Device Manager**→**Database Utilities**→**Database Verify Repair** from the Windows taskbar).
- If the event section of the merge file is empty, the time range specified for the export did not include the correct times and dates. Try exporting again, making sure that the date and time when you made the template is included.

Installing and Testing Your Device Files

This section explains how to install your device into an existing AMS Device Manager system to test your device interface.

Hardware You Will Use

You will need the following hardware to test your device:

- At least one available PROFIBUS Interface - To connect your device to AMS Device Manager.
- Your device, connected to AMS Device Manager.

Assumptions

This section assumes that:

- You have installed AMS Device Manager.
If you need more information about installing AMS Device Manager or selecting a data source, see the *AMS Suite: Intelligent Device Manager Installation Guide*.
- You have created a Device Installation Kit.

Overview of Device Installation and Test Steps

Step T-1: Configure Your AMS Device Manager Network and Connect Your Device

After you have created your device installation kit, you are ready to connect your device to AMS Device Manager in order to test your device.

- Refer to the *AMS Suite: Intelligent Device Manager Installation Guide* for information on how to connect your device to AMS Device Manager.

Step T-2: Start AMS Device Manager and Test Your Device DLL

Once your device is connected to AMS Device Manager, test the functionality of your device:

1. Start AMS Device Manager (refer to the AMS Device Manager Books Online if you need instructions).
2. Open the Device Explorer View, if it is not already open (**View**→**Device Explorer View**). You should see your device attached to the Interface.

Note: *The first time you run AMS Device Manager with your device connected, there will be a short delay as AMS Device Manager initializes its database with your device information.*

3. Select your PROFIBUS Interface and perform a Rebuild Hierarchy operation.

4. Perform a Scan New Devices operation.
5. Double-click your device icon to open the faceplate and verify that it looks as intended.
6. Right-click on the faceplate to open the context menu. Verify that the context menu looks as intended. Verify the operation of all context menu items.
7. Verify all of the methods available from the context menu as needed and test each method, ensuring that they complete without errors.
8. Verify that the Status screen appears as intended. Launch any other context menu items. Configure device in Alert Monitor. If possible, cause your device to report an error status and verify that the correct status is displayed.
9. Verify that the Configuration Properties screen looks as intended. Verify that each parameter intended to be editable from the Configuration Properties screen is editable. Ensure that all writable values on the screen actually write to the device successfully when you select **Apply**. Verify labels, tab order, and parameter Help. Repeat the following steps for each block of the device:
 - a. Read each configuration parameter (current, history).
 - b. Verify that the data appears to be valid (e.g., no * are in any fields).
 - c. Write each configuration parameter (current).
 - d. Reopen and verify the newly written parameters.
 - e. Verify PRE_EDIT_ACTIONS, POST_EDIT_ACTIONS, PRE_READ_ACTIONS, POST_READ_ACTIONS or PRE_WRITE_ACTIONS, if applicable.
10. Verify refresh relations and unit relations, if applicable.
11. Ensure that all writable values on the screen actually write to the device successfully when you select **Apply**. Verify that the Compare Configuration screen looks and functions as intended. Verify that each parameter intended to be editable from the Compare screen is editable. Verify labels, tab order, and parameter Help. Perform the following steps for the device:
 - a. Read each current, history configuration data for each parameter (source and destination).

- b. Verify that the data appears to be valid (e.g., no * are in any fields).
 - c. Transfer config data.
 - d. Write any current configuration parameter (source and destination).
 - e. Reopen and verify the newly written parameters.
 - f. Verify PRE_EDIT_ACTIONS, POST_EDIT_ACTIONS, PRE_READ_ACTIONS, POST_READ_ACTIONS or PRE_WRITE_ACTIONS, if applicable.
12. Verify refresh relations and unit relations, if applicable.
 13. Verify that an offline configuration can be created and edited correctly.
 14. Verify that all Help strings are displayed as intended.
 15. For each mode, if applicable, verify that all displays look and function as intended.
 16. For each mappable dynamic variable, if applicable, verify that all displays look and function as intended.
 17. Execute Diagnostic and Test methods multiple times using multiple paths. Verify that the text on the dialogs displays completely and that the method completes without error.
 18. Execute Calibration Methods multiple times using multiple paths. Verify that the text on the dialogs displays completely and that the method completes without error.
 19. Execute a Master Reset and verify the proper dialog display and test completion.
 20. Verify that Process Variable parameters can be read, updated, and will launch other context items.
 21. Verify that you can change the name of the device.
 22. Verify that the drawings/notes for the device can be entered and saved correctly.
 23. Verify that entries are being created correctly in the Audit Trail. Select an individual entry and verify that multiple values are defined as having been changed.
 24. Verify the ability to switch between modes of Multi-Mode Devices.
 25. Verify Winhelp operation:

- a. Make sure the device has a .CHM reference in the FMS.INI in the BlockHelp parameter.
- b. Launch the device Configuration, Process Variables or Status/Conditions screen.
- c. Select the “What's this” help button and click a field on a tab.
- d. Verify the WinHelp topic appears.

Note: *These steps for testing are not meant to be all-inclusive, but only to be a guideline for testing your device DLL.*

Step T-3: Modify Your Device Installation Kit

If you discover a problem with your device in Step T-3, edit the file to fix the problem and rebuild your device installation kit.

Step T-4: Copy Your New File into the AMS Directory

Once you have modified the file and rebuilt your device installation kit, test your changes by copying the file into the device installation kit you created and rerun Add Device Type.

Step T-5: Repeat Steps T-2 through T-4 as Needed

While you are testing your device, you may decide to make some changes. If so, repeat Steps T-2 through T-4 until you are satisfied with the functioning of your device with AMS Device Manager.

As part of this testing, you may need to modify the DD for your device. If so, make the modifications, tokenize the DD, and copy the PBO and SYM files into the proper ...\\DEVICES\\PROFIBUS subdirectory.

Note: *Do not Drag and Drop files in to the AMS\\Devices Folder. Instead, use the Copy and Paste function.*

The following paragraphs contain some additional, specific testing to perform on your devices, if applicable.

Log Files for Debugging Your Device

To log the traffic between your PROFIBUS server and the DeltaV Host system, add the following entry in the DeltaV network section of the FMS.INI file:

```
DVPBUSTRACECOMM=<filename>
```

Ensure that you enter a valid filename. Once this entry is added, the DeltaV network section of the FMS.INI file should look like this:

```
[DeltaV Network 1]  
Name=DeltaV Network 1  
DVPBUSTRACECOMM=c:\ams\Comm.log  
DeltaVComm Server=ams-dev1  
User name=DeltaVAdmin
```

Clean Database

Make a backup of your AMS Device Manager database before you add your device. Later you can restore the database without the device, if necessary.

Integrating the Device into AMS Device Manager

This section explains how to integrate your device into AMS Device Manager once the installation kit has been created.

Methods for Integrating into AMS Device Manager

There are two methods for integrating the device into AMS Device Manager. The first is through an independent integration which calls for distribution by your company of the installation kit for the device created in the previous section and sending this kit directly to the customer. The second method is through a full integration, whereby the device is distributed with AMS Device Manager to all customers automatically.

Independent Integration into AMS Device Manager

Using the Device Installation Kit created in the previous section, distribute this disk to customers requiring the device operation in AMS Device Manager.

This method is also useful when putting out modifications or new releases of devices that do not coincide with a scheduled release of AMS Device Manager.

Additionally, Emerson Process Management has a web site to provide a convenient location to download Device Installation Kits, and for device developers to make updates available more quickly to their customers. The web site can be found at:

<http://www.emersonprocess.com/ams/prodedes.htm>

You can email questions about this site to:
amsdevicemanagertoolkit@emerson.com.

**Full Integration into
AMS Device Manager**

This method allows the device to be included with the release of AMS Device Manager to all customers. The AMS Device Manager build process requires all the files that are normally distributed with the installation kit.

Device Description .PBO

This represents the tokenized DD Binary all source files required to tokenize the device description to the .PBO files, including .DD, .H, and any other device-specific files imported into the device description.

Device Merge File

The device merge file, created in the previous section, contains a default template for the device as well as information about the manufacturer and device.

DDINSTALL.INI

This file shows AMS Device Manager how to install the device. It is incorporated into the overall install file with all other integrated devices.

The Integration Timeline

Device installation kits can be submitted at any time to amsdevicemanagertoolkit@emerson.com. They will be integrated into the next release of AMS Device Manager and placed on the AMS Device Manager DD download website.

The Next Step

If you have questions about the integration process or any issues with the integration of your device into AMS Device Manager, send an email to: amsdevicemanagertoolkit@emerson.com.

AMS Device Manager Device Context-Sensitive Help-getting help beyond DD help strings

This appendix contains a description of the procedure for creating context-sensitive “Advanced” device help. Also included is information about providing a product manual in .PDF format.

Introduction

This section shows device developers how to create device-based online help that extends beyond textual help provided in the Device Description. As of AMS Device Manager 10.0, this help may consist of:

- Advanced Help for a device parameter is displayed by clicking the **More** button within a DD help window. See “Providing Advanced Help”.
- A Product Manual, provided in .PDF format. See “Providing a Product Manual” for an example.

In addition to this appendix, there are sample files provided to guide you through creating Advanced Help, and providing a .PDF file. See the folder User Assistance for sample files you may want to use in creating your system.

Providing Advanced Help

Advanced Help for EDD devices involves information beyond the standard EDD Help string. Beginning in version 8.x, you can provide Advanced Help by connecting parameter names of VARIABLE, GRAPH, CHART, and IMAGE to topics in your Help file.

Advanced Help for devices is in context to the DD Help window that a user launches, and can be done in HTMLHelp

Assumptions

AMS Device Manager requires that a Help attribute be defined in the EDD for every parameter for which you provide Advanced Help. A user links to Advanced Help from the EDD Help window.

You can provide one .CHM file per Device revision. **HTMLHelp** HTMLHelp is a set of HTML topics compiled into a binary format (.CHM). These compiled files are viewable on any Windows PC with Internet Explorer. The end user does not need to install or associate any applications to view the file.

To create a CHM file for your device, you must have Microsoft HTMLHelp Workshop. Microsoft HTMLHelp Workshop is the free Windows Help compiler available from Microsoft at:

<http://www.microsoft.com/downloads/details.aspx?FamilyID=00535334-c8a6-452f-9aa0-d597d16580cc&DisplayLang=en>

Although you may find it helpful to purchase a third-party tool for creating HTMLHelp, it can be done with an HTML editor and Microsoft HTMLHelp Workshop alone. A very inexpensive tool that handles .HHP and .HHC editing better than the compiler is called FAR. It can be purchased from www.helpware.net.

The source files for an HTMLHelp system shall consist of the following:

- The project file (.HHP) – The project file contains the settings for the Help system.
- HTML file(s) – The content.

The <device.ini> file – The file that maps the parameter names to the HelpIDs. The files you deliver as part of your DDinstal files consist of the following:

- The .CHM file – The compiled HTMLHelp file. It must be where the application expects it.

The process for creating an HTMLHelp system for your device is as follows. Each step below will be described in more detail:

1. Determine which EDD parameters will need Advanced Help.
2. Create the [ADVANCED HELP] section of the <device.ini> file.
3. Create the map file
4. Create the project file.
5. Create the topics.
6. Create the table of contents.
7. Create the index.
8. Compile the project.
9. Test the project.
10. Add the Help files to the install kit.

Before you start creating your Help system, you must decide on a name for it. This is the name that will be used for source files and the files that are delivered to the user. A good nomenclature is <devicetype><devicerev><ddrev>.hlp.

1) Determine which EDD parameters will need Advanced Help

2) Create the device.ini file [ADVANCED HELP] section

```
[AdvancedHelp]
buffer_standard=16894
device_status=151
glass_high_fault=16901
glass_low_fault=16902
hold_config=16873
```

3) Create the map file

Copy the values defined in the ADVANCED HELP section of the <device>.ini to a <device>.h file and reference it in your project (see [MAP] section of sample .HHP file).

The HTML Help compiler expects a format nearly identical to the [ADVANCED HELP] section. Simply replace the = sign in with a space, and your IDs will be mapped.

Create the project file

11. See the sample HHP file. Create a [MAP] section to connect the project file to your device.h file:

```
[MAP]
```

```
#include <device>.h
```

If you compile the project with the #include statement above, the device.h file must be in the same folder as the .HHP. Create an [ALIAS] section, and map the Topic ID to the .htm file with the following format:

```
<DD parameter name>=<path to .htm file>;<comment>
```

```
HID_DEVICE_WINDOW_CONFIGURE_VIEW=DEVCONF_OVER_ENHANCED  
_DEVICE_WINDOW.htm ; Configure view
```

This section assumes the htm files are in the same folder as the .HHP.

For more information on the project file settings, see the Microsoft HTMLHelp Workshop online help.

Create the topics

Create a folder for your project, and then launch your HTML editor and create the HTML file(s) that will contain your help topics.

Your files must be valid HTML 4.0 or valid XHTML.

You will be creating one topic per device parameter.

See the online help in Microsoft HTMLHelp Workshop or the following MSDN articles for more information on creating topics:

[http://msdn2.microsoft.com/en-us/library/ms670169\(VS.85\).aspx](http://msdn2.microsoft.com/en-us/library/ms670169(VS.85).aspx)

<http://msdn2.microsoft.com/en-us/library/ms669980.aspx>

Create the table of contents

To create a table of contents that appears on the Contents tab in the Help viewer, you can use the HTMLHelp Workshop HHC editor. There are also 3rd party tools available for this.

You will need to integrate your Table of Contents inside the AMS Device Manager Table of Contents. See the sample file for the correct structure of the .HHC file.

Create the index

To create an index that appears on the Index tab in the Help viewer, you can use the HTMLHelp Workshop HHK editor. There are also 3rd party tools available for this. By default, keywords are maintained in a separate file, but you can also use HTML <META> tags to keep the keywords in the file they describe. When editing the project file (.HHP), select the Change Project Options button on the top left, and select the option "include keywords from HTML files".

Follow these guidelines when you create the index:

- Use the terminology guidelines developed by Emerson Process Management Common Terminology Team.
- Check the index of ams.chm to determine if your index term is being used consistent with that document's index.
- Index to no more than two levels.
- Avoid single secondary keywords.
- Only capitalize the first word of an index entry if it is a proper name.

Compile

To create the .CHM file, you compile your Help system. To compile your help system, click  in Microsoft HTMLHelp Workshop. If you have set the project file to display notes, you will see a list of messages about the compilation. Look at the messages to see if they indicate errors that you need to address.

Test

After successfully compiling the file, copy it to the proper device folder and then test it in AMS Device Manager.

1. Modify your Device Installation Kit.
2. Copy the .CHM to the installation kit folder.
3. Update the DD install.
4. Device help = YourHelp.CHM

5. Run the Add Device Type utility to add the files to the system.

You are responsible for testing the content and integrity of your help system. A Unit Test Plan included in the sample will give you some ideas about how to test a compiled HTML Help file. It is not an exhaustive list.

AMS DEVICE MANAGER IS NOT RESPONSIBLE FOR TESTING THE CONTENT OF YOUR HELP FILE.

The parameters with help should display the DD help with a “More” button. Clicking on this button will bring up your help file in context.

Add the help files to your device folder

When the Help for the device is complete, add it to your device file shipment by copying the .CHM into the folder that contains the rest of the device files. When we know you are providing Advanced Help, we will modify AMS Device Manager to enable launching of the files. Make sure that your files are included in your device integration checkout process or process approval form.

Writing Guidelines

Conventions

The following writing conventions should be used to ensure consistency with existing product online Help.

Design-test your device’s UI with multiple users outside your development organization. A good UI can alleviate the need for extensive Advanced Help.

Content

Follow these guidelines when writing the content for context-sensitive (More button) help:

- The user’s primary help for your device should be a .PDF or DD-based help. Making the user click on a More button should only be done if the explanation for the parameter in DD help is insufficient.
- The potential values/entries should be explained in the context of the mode of the device so the user can be guided towards making the right configuration settings. Explain what fields are required to be addressed by the user and which ones are not.
- Provide troubleshooting sequences, recommended actions for alerts, and practical advice for the technician/engineer when appropriate.

- Where appropriate, use graphics, and links to other topics, See the Sample Help System provided with this toolkit for instructions on how to include these in your help topics.

Writing Style and Usage

Follow these styles and rules when writing your content:

- Use active voice.
- Use present tense wherever possible. Future tense is appropriate occasionally, but in most cases it is not.
- Use second person ("you") rather than the third person ("the user") wherever possible.
- Be as concise as possible, using only as many words as needed to get the meaning across. For example, use: "lets you" instead of "allows you to"; "To..." instead of "If you want to..."
- Use an informal—but not chatty—writing tone.
- Do not use contractions.
- Use “click” not “click on” to designate selecting a button using the mouse (“Click Help”) except for something like “by clicking once on its label”.
- Use login as a noun form “the login screen” vs. log in as a verb “log in to AMS Device Manager”. Do not hyphenate.
- Use "on a tab" vs. "in a tab."
- Use "want" vs. "wish."
- Use "PC" or "computer" vs. "machine."
- Use "window" vs. "screen" when referring to a view in an application.
- Use "dialog box" when referring to the type of window that elicits a response or input from a user.
- Lists:
 - Introduce a list with a complete sentence. This aids in translation efforts.
 - Use unordered (bulleted) lists for lists of items.
 - Use numbered lists for first-level procedures and lettered lists for second-level procedures.
 - Make items in a list parallel.
 - Capitalize the first word of each item in a list.

- End each entry with a period if all entries are complete sentences, are a mixture of fragments and sentences, or complete the introductory sentence. An exception is when all entries are short imperative sentences (three words or fewer) or single words; these entries do not need a period. If all entries are fragments that do not complete the introduction, do not end them with periods.
- Use hyphens (-) vs. em dashes (—) to explain items in a list (for example, "AMS Tag - The AMS tag for the device").
- Notes:
 - Use notes sparingly. Use only for exception or aside information. Do not overuse notes, or they lose their value.
 - In HTML Help topics, do not use multiple notes under one "Notes" heading. Multiple notes must be implemented as separate back-to-back notes. Avoid back-to-back notes if at all possible. If back-to-back notes are absolutely necessary, avoid using more than two.
- Titles:
 - Use the imperative mood for the verb in a procedure topic title in HTML Help ("View the Audit Trail for an AMS tag", "Scan devices").
 - When an overview topic is about an action, use the gerund form of the verb in the title ("Importing and exporting data to and from the ValveLink standalone application", "Troubleshooting Drawings/Notes").
 - Use "Subject " instead of "About subject" in an overview topic title, except in titles that use gerunds. Never use the word 'Overview' in a title with a gerund (for example "Replacing Assigned devices"). Avoid using "Subject Overview" unless there is enough information about a subject to warrant multiple reference topics. In that case the Overview topic is that subject's "home page" and the destination for links. In titles with gerunds, do not use the word "overview" (for example, "Replacing Assigned devices").
- When the text refers specifically to words or numbers that appear on a display, the text should exactly duplicate the capitalization and punctuation as it appears on the display. For example: "In the Configuration window, select Offline in the Time field." In this example the word "Offline" is capitalized because it is capitalized on the display. But in the following

example, the word "offline" is not capitalized, because it is used in a generic sense: "Only one offline configuration can be associated ..."

- Use "offline" and "online" as single words, not hyphenated.
- Use startup as a noun ("faster startup") and start up as a verb ("start up User Manager"). Avoid the ambiguous hyphenated form "start-up".
- Indicate use of a key on the keyboard by the word "press" (for example: "... press ENTER to commit the changes"). Do not say "... press the ENTER key ..." or "... press down the ENTER key."
- Numbers:
 - Common practice suggests numbers between zero and nine be written out, presumably to increase their readability. Invariably, the best rule to follow is "watch the context".
 - Be consistent when dealing with multiple sets of numbers in a single sentence.
 - When describing the User Interface, use what the UI uses.
 - Technical documentation, with its focus on clarity, often relies on numerals for its precision. Numerals should almost always be used when defining ranges, indicating date or time, capacity (with units), or default settings.

Examples:

This option lets you select four types of database problems.

This value must be an integer in the range 1 through 14.

A minimum of 5 test points for one direction or 9 points for two directions are required to calculate zero, span, or linearity.

Writing for Translation

There are many resources, publications, and associations available to assist in writing for translation. The Globalization and Localization Association (GALA) is one well-known association. Improving translatability in the English document avoids repetitive costs translating and fixing problems for each language. This list is a small portion of issues that frequently arise. Consult your translation or localization vendor prior to beginning an effort for further ideas.

Writing Tips

- Avoid using the construction “(s)” to indicate “either singularity or plurality”. This usually translates into poorly-constructed non-English sentences, and adds cost to translation. Whenever possible, choose either the singular or plural. (ex. Poor: “Enter the point(s) to complete the transaction.” Better: “Enter any points to complete the transaction.”)
- Strive to introduce bulleted lists with complete sentences.
- Develop a structured language with terms that are allowed. Avoid using synonyms whenever possible to improve consistency.
- Avoid ambiguity, such as the following: “Once charging ends, unplug the unit.” In this context “once” could mean “when”, “after” or “if”.
- When describing amounts or units, use the International System of Units (SI). It is a standardized list of acceptable abbreviations. If possible, and when supported by your text editor, use text entities for units rather than combinations of letters.
- Avoid unnecessary and difficult words. (For example, use “total”, not “aggregate”)
- Be consistent and reuse blocks of text.
- Globalize references to date, time, address, and phone numbers.
- Avoid subordinate clauses.
- Avoid nouns created from verbs. (For example, “refuse” vs. “refusal”)
- Match names of menus, buttons, and other UI labels exactly (or as closely as possible) in the documentation.

Layout and Design Tips

- Separate graphics from text.
- Avoid changes to the format or text for stylistic reasons.
- Clean up formatting (such as extra carriage returns) and use styles.
- Provide white space for tables and text to expand. Translated text will expand by about 30 percent.
- Be aware of color theme preferences by region.

- Use common fonts.

Abbreviations and Acronyms

- Define acronyms in the glossary.
- In HTML Help topics, do not spell out the first occurrence of an abbreviation or acronym in a topic. Instead, link the first occurrence to its definition in the glossary, but only if the abbreviation or acronym is one that may not be familiar to some users. For example, do not link "HART" to its definition in the glossary, because HART is a well-known term.

In user manuals, show the full spelling of an abbreviation or acronym the first time it is used (put the spelled-out form first, followed by the abbreviation in parentheses).

- Do not use Latin abbreviations: do not use i.e., e.g., or etc.; instead, spell out the words "that is," "for example," "and so forth."
- Do not use "via." Use "by," "through," or "by means of" instead.
- Spell out any other abbreviations that might be misunderstood by the reader.
- Spell out symbols such as "percent" and "dollars" rather than using % and \$ in text. However, use symbols where there is a space limitation, as in tables or figures.
- Form the plural of an abbreviation by the addition of a lowercase s. Do not use 's, unless the meaning might be confused or the abbreviation contains periods.
- Use "FF" as the abbreviation for FOUNDATION fieldbus only when "FF" is used in the user interface, and even those occurrences should be used sparingly.

Capitalization

- Capitalize proper names. Proper names include names of objects in the user interface, such as icons, windows, dialog boxes, fields, and buttons; and pieces of functionality (such as Alert Monitor or Audit Trail). Leave the word for the object ("window", "dialog box") in lowercase (for example, "Compare Configurations window", "Export dialog box").

The decisions about what are proper names are reflected in the glossary for the product. (The decisions may sometimes seem arbitrary.) If you are unsure about whether an item is a proper name, you should consult with other members of the writing team.

- Capitalization of an item in the UI does not necessarily mean the item is a proper name.
- Terms used in their “verb form” are not capitalized. (For example, "The alert monitoring...".)
- Capitalize only the first word and proper names in headings (all levels)
- Capitalize only the first word and proper names in HTML Help topic titles (the <title> attribute) and in Related Topics links.
- In user manuals, capitalize the first letter of significant words in figure and table titles, section titles, and headings.
- When referring to online Help, the word "Help" is capitalized.
- Full-cap HART (as in HART Communication Foundation, HART protocol, or HART device). HART is an acronym for Highway Addressable Remote Transducer.
- Full-cap and hyphenate SNAP-ON. Always follow the term SNAP-ON by the term “application,” for legal reasons.
- When referencing another section of a user manual, initial-cap the word "Section".
- Use "FOUNDATION fieldbus" in high-level topics and "fieldbus" everywhere else. Use "fieldbus" and "FOUNDATION fieldbus" interchangeably, with the following standardization:
 - "Fieldbus" is capitalized only when it is part of the full title "Fieldbus Foundation."
 - When the usage is "FOUNDATION fieldbus," or when the word "fieldbus" is used alone, it is never capitalized.
 - "FOUNDATION" is initial capitalized AND written in all small caps when the usage is "FOUNDATION fieldbus."
 - When the usage is "Fieldbus Foundation," both words are capitalized, but the small caps are not used for the word "Foundation."
- Avoid the abbreviation FF in most cases. Spell out FF as “FOUNDATION fieldbus” (use all caps if small caps is not available in your font).

Punctuation

- Use one space after a period.
- Use hyphens, en dashes, and em dashes appropriately (follow the model already in use).

- When a singular noun ends in 'ss', and it needs possession, ss' is preferred over ss's (example boss', business').
- Use commas to separate ALL items in a series (for example, "device configurations, user configurations, and device templates").

Example of MORE button

1. User clicks the What's This? button in AMS Device Manager toolbar and then clicks on a parameter.
2. If Advanced Help is available, the More button is displayed as shown in Figure 5-1.
3. When the user clicks the More button, the associated Advanced Help is displayed.

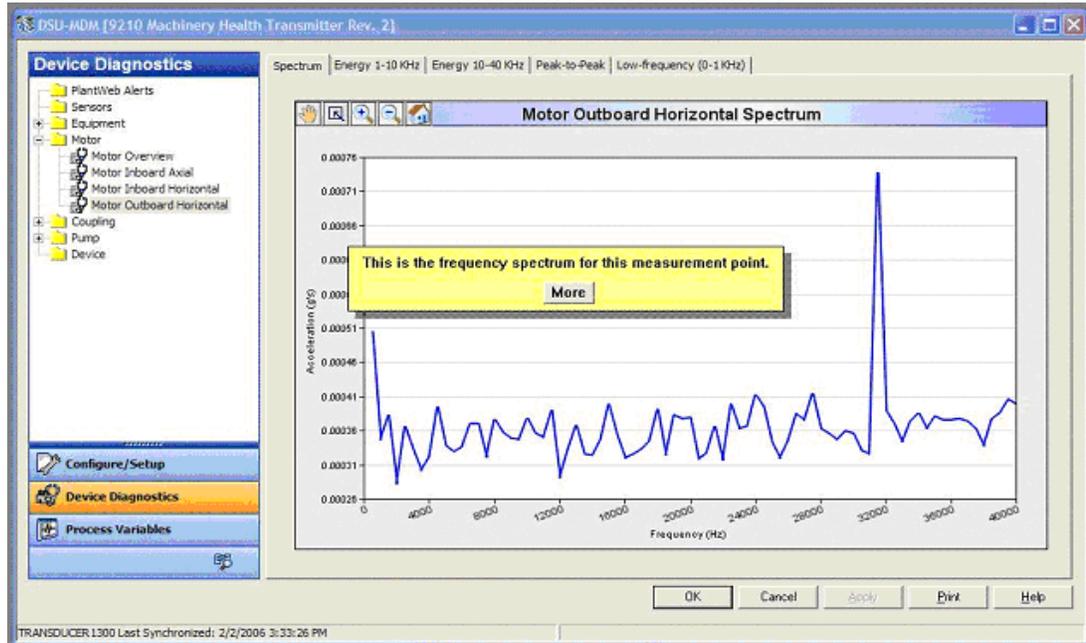


Figure 5-1. Example of a More button

Providing a Product Manual

You should include a product manual in PDF format with the AMS Device Manager shipment. The product manual is accessed from two locations in AMS Device Manager:

- From the Help | Device submenu, where a user selects your device from a list.
- From the Help selection on your device's context menu.

You may send a single product manual for each device description revision you are supporting in AMS Device Manager, or you may ship one manual that opens for each device. If you choose not to provide a product manual, your device will not appear in the Help | Device submenu, and a generic topic from AMS Device Manager will display when the user chooses the Help selection on the device context menu.

Your product manual should follow the guidelines listed in “Writing Guidelines”, and should clearly indicate the device, device revision, and DD revision (if applicable). Emerson devices should follow appropriate branding guidelines. The name for your Product Manual in AMS Device Manager should indicate the device(s) to which it refers. A good nomenclature is <devicetype><devicerev><ddrev>.PDF.

Add the PDF to your device files folder

Add your .PDF to your device file shipment by copying the .PDF file into the folder that contains the rest of the device files. When we know you are providing a product manual, we will modify AMS Device Manager to enable launching of the files. Make sure that your files are included in your device integration checkout process or process approval form.

Test

Modify your device's DDINSTALL.INI file to Device_Help=<yourfilename.PDF>. This allows AMS Device Manager to launch the PDF from the Help | Device submenu, as well as the Help selection on your device's context menu.

AMS DEVICE MANAGER IS NOT RESPONSIBLE FOR TESTING THE CONTENT OF YOUR PRODUCT MANUAL FILE.