



Digi Device Customization and Integration Guide

Digi Connect Family products
Digi Connect ES Family products
Digi Cellular Family products
ConnectPort™ Display products
ConnectPort X Family products
ConnectPort X5 Family products
ConnectPort TS products
ConnectCore 3G 9P 9215

www.digi.com

©2010 Digi International Inc. All Rights Reserved.

The Digi logo, Digi Connect, ConnectPort, ConnectCore, and RealPort are trademarks or registered trademarks of Digi International, Inc. in the United States and other countries worldwide.

All other trademarks are the property of their respective owners. Information in this document is subject to change without notice and does not represent a commitment on the part of Digi International.

Digi provides this document “as is,” without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Digi may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes may be incorporated in new editions of the publication.

About This Guide	5
Purpose	5
Scope	5
Audience.....	5
Where to Find More Information.....	6
Digi Contact Information	6
Chapter 1 Introduction	7
Customization.....	7
How much customization does your device require?	7
Integration.....	7
Customization and integration tools and interfaces.....	9
Product look-and-feel customization tools.....	9
Manufacturability resources.....	10
Software interfaces and redeployable elements.....	10
Hardware interfaces	11
Customization and integration scenarios.....	11
Customize Digi device look-and-feel	11
Manufacturability	12
Implement software interfaces and redeploy elements	12
Implement hardware interfaces	12
Chapter 2 Tools and Interfaces	13
Product look-and-feel customization tools.....	13
Web interface customization tools	13
Custom factory defaults.....	21
Configuration applet	22
Manufacturability tools and resources	24
Load preferred firmware image	24
Digi Connect Programmer	25
Sample manufacturing environment.....	26
Software interfaces and redeployable elements.....	28
Advanced Digi Discovery Protocol (ADDP) - Device discovery program	28
Remote Command Interface (RCI)	29
Simple Network Management Protocol (SNMP).....	30
RealPort drivers	32
Digi device product documentation	32
Hardware interfaces	33
Reset functions.....	33
RCI over serial.....	33
Chapter 3 Digi Connect Programmer	34
Start Digi Connect Programmer	35

General usage information	36
Digi Connect Programmer parameters.....	37
For specifying the target device.....	37
For configuring network settings.....	37
For uploading Web customization and configuration files	37
For uploading file systems.....	37
For uploading firmware files	37
Parameters used as flags.....	38
/discover - Discover devices.....	39
/set - Set network parameters	40
/upload - Upload files to devices.....	41
/defaults - Revert device to factory defaults	45
/reboot - Reboot device	46
/? and /help - Display help text	47
Digi Connect Programmer error messages	48
Digi Connect Programmer error codes.....	49
Chapter 4 Scenarios	50
Customizing Digi device look-and-feel	50
How do I customize and save changes to the web interface?.....	50
How do I upload the customized web interface to other devices?.....	51
How do I change the customization username and password?	52
How do I change default settings in a device (custom factory defaults)?	53
How do the web interface and applet differ? Which should I use?	55
How do I create an applet?.....	56
How do I upload an applet?	57
How do I revert a customized device to the Digi-assigned defaults?	58
Manufacturability	59
How can I efficiently configure and deploy devices?	59
How do I upload a customized web interface to other devices?.....	59
How do I upload custom factory defaults to multiple devices?	59
How do I upload an applet to multiple devices?	59
How do I load a preferred firmware image during manufacturing?.....	59
What do I need to set up a manufacturing environment?	60
Implementing software interfaces.....	61
What software interfaces are available on Digi devices?	61
How does device discovery work, and how can it be modified?.....	61
What is the Remote Command Interface (RCI)?	63
Can I configure and monitor devices using SNMP?	63
Can I redeploy RealPort drivers?	63
How can I customize the product documentation?	64
Implementing hardware interfaces	65
How should I implement reset functions on my Digi device?.....	65
How can I use RCI over serial?	66
Index.....	67

About This Guide

Purpose

The purposes of this guide are to:

- Define the concepts of customization and integration in relation to Digi devices.
- Introduce the tools and resources available for customization and integration.
- By scenarios and examples, show how to customize Digi devices and integrate the customized elements into your own products and solutions.

Scope

This guide applies to customization and integration efforts for these Digi product families running plug-and-play firmware:

- Digi Connect Family products
- Digi Connect ES Family products
- Digi Cellular Family products
- ConnectPort™ Display products
- ConnectPort X Family products
- ConnectPort X5 Family products
- ConnectPort TS Family products
- ConnectCore 3G 9P 9215

It is not intended to teach all general elements of customization and integration. Where such general elements are involved, it refers to publicly available Web sites and articles for more detail; for example, the basics of stylesheets and their application.

This guide focuses on tools and processes involved in software integration; that is, how to discover and configure the device, manufacture the device, and the software involved in these processes. This guide also discusses how to customize an NDS module to operate with a custom baseboard. A more detailed explanation of hardware-related integration is covered in the *Hardware Reference Manuals* for Digi products, if available.

Audience

This guide is intended for software and hardware engineers at a company who are responsible for configuring and customizing devices to their own company's specifications before the device is released to their customers/clients, and for integrating Digi devices into their company's products.

Where to Find More Information

Other documents that are of assistance in customization and integration efforts include:

- *Remote Command Interface (RCI) Specification*: describes the data elements of the Remote Command Interface protocol that can be used as an Extensible Markup Language (XML) programmatic interface to Digi devices.
- *User's Guide* for the Digi device's product family: describes the standard features, functions, and interfaces for Digi devices.
- *Digi Connect Family Command Reference*: describes the standard command-line interface for Digi devices.
- Hardware Reference manuals for Digi devices, if available. Several Digi devices have manuals describing hardware characteristics the products, and the capabilities and programming considerations of these modules in integration efforts. If a Hardware Reference is not supplied, a short summary of the hardware specifications is provided at the end of the *User's Guide*.
- Sample applications and accompanying documentation provided with the Digi Connect Integration Kit.

Digi Contact Information

To contact Digi International for more information about your Digi products, or for customer service and technical support, use the following contact information:

To Contact Digi International by:	Use:
Mail	Digi International 11001 Bren Road East Minnetonka, MN 55343 U.S.A.
World Wide Web:	http://www.digi.com/support/
email	http://www.digi.com/support/eservice
Telephone (U.S.)	(952) 912-3444 or (877) 912-3444
Telephone (other locations)	+1 (952) 912-3444 or (877) 912-3444

Customization

Customization is the ability to private-label and customize a Digi device. Several aspects of Digi devices can be customized; for example:

- The company and product information displayed in device user interfaces can be changed.
- The look-and-feel of the device interface for Digi devices can be customized, to use a different company logo or screen colors.
- A set of custom factory defaults can be defined, allowing you to run a device using your own settings, and as needed, revert back to your custom factory default settings and not the Digi default settings.
- A preferred firmware image can be loaded into the Digi device, for example, a firmware version level that a company prefers to use as a standard.
- Custom Java applets can be created, using the Java configuration applet as a sample for further development.
- Customize a Digi Module for a specific baseboard design (see Chapter 5, "NDS Module Customization").

Such customization allows you to release and distribute Digi devices as your own product. The amount and extent of the customization depends on the requirements of your company.

The Digi Connect Integration Kit provides a platform for evaluation, rapid prototyping, and integration of Digi devices with plug-and-play firmware. It includes tools, sample code, and documentation to help with your customization efforts. Chapter 2, "Tools and Interfaces" describes these customization resources.

How much customization does your device require?

As part of the customization effort, you need to define the scope of customization needed for your device. For example:

- Do different colors need to be displayed in the device's user interfaces?
- Does the product need to have a different name, or appear as a completely different product?
- Are other look-and-feel changes needed for the user interfaces?
- Do the factory default settings need to change to a set of defaults specific to your company? If so, how many settings and which ones?
- Does a particular firmware image need to be loaded into the device?

Integration

In general, integration involves how a Digi device is used in your company's product, service, or solution.

There are two aspects to integrating a Digi device:

- Hardware integration, which involves the physical characteristics of the Digi product. These physical characteristics are covered in the *Hardware Reference Manuals* for Digi devices.
- Software integration; that is, how you discover and configure the device, manufacture the device, and the software involved in these processes. Tools and processes for software integration are covered in this Guide.

Examples of integrating a Digi device into a product, service, or solution include:

- Building access control equipment
- Bar-code reader
- RFID reader
- Medical equipment
- Power management

Customization and integration tools and interfaces

Several tools and interfaces are available for customizing Digi devices and integration efforts. Chapter 2, "Tools and Interfaces" discusses them in more detail.

Product look-and-feel customization tools

Tools that help you customize the product look-and-feel of the Digi device user interface and device configuration interface include:

- The *web interface* for most Digi devices has a **Customization Administration** page. This page displays the history of customization actions for the device. You can download and upload files used for customizing the product look-and-feel, such as the configuration file, custom factory defaults, and stylesheet. For multi-user Digi devices, this page can be password-protected from end-user access. This page is described in more detail on page 14.
- A configuration file for the web interface, named **config.ini**, for customizing identity-related content in the user interface. You can use this file to customize such things as the product name, company name, and logo displayed on the device user interface. This file is described in more detail on page 16.
- A Custom Factory Defaults Template, named **defaults.rci**, for customizing the default settings for the Digi device. This template is described in more detail on page 21.
- A stylesheet template, which can be any filename but must have the extension **.css**. The stylesheet controls the look and operation of the Web device interface. This template is described in more detail on page 21.
- The Java configuration applet, an alternate user interface that can be completely customized to your needs. This applet is described in more detail on page 22.

Manufacturability resources

Once a Digi device has been customized to your needs, the next step is replicating the customized user interface, device settings, and, if needed, a particular level of firmware onto multiple Digi devices. The equipment and processes involved in configuring and deploying Digi devices are known as manufacturability resources.

The Digi Connect Integration Kit includes several manufacturability resources to aid you in efficiently configuring and deploying devices, and in setting up a manufacturing environment.

- A preferred firmware image can be loaded into Digi devices, as described on pages 24 and 59.
- The Digi Connect Programmer, a command-line interface, is available for efficiently loading custom factory defaults and other customization files into Digi devices. The Digi Connect Programmer supports commands to perform a number of other operations, but its primary use from a customization and integration standpoint is efficient file loading. "Digi Connect Programmer" on page 25 and "What do I need to set up a manufacturing environment?" on page 60.
- A sample manufacturing environment is provided. This sample demonstrates the components involved in configuring and deploying multiple Digi devices, including scripts of Digi Connect Programmer commands and other commands used to configure Digi devices. You can build upon and modify this sample to set up your own manufacturing environment. See pages 26 and 60 for more information.

Software interfaces and redeployable elements

Digi devices support a number of common interface protocols, such as Telnet, raw TCP sockets, UDP sockets, RealPort, HTML, and others. In addition to these common interfaces, there are several Digi International-proprietary interfaces, or implementations of a Digi-specific programming elements:

- Advanced Digi Discovery Protocol (ADDP), a Digi International-proprietary protocol for discovering devices on networks. See page 28.
- Remote Command Interface (RCI), a programmatic interface for configuring and controlling Digi devices. See page 29.
- Device configuration and monitoring through Simple Network Management Protocol (SNMP) involves both standard Management Information Blocks (MIBs) and several Digi-specific MIBs.

Also, there are elements of the standard Digi devices that can be redeployed in your company's software distribution, including:

- RealPort drivers; see pages 32 and 63.
- Digi device product documentation; see pages 32 and 64.

Hardware interfaces

Several hardware-related features are available for use in your customization and integration efforts.

- Reset functions, including hard and soft reset, can be implemented on Digi devices. See pages 33 and 65.
- The Remote Command Interface (RCI) protocol can be used over serial as well as Ethernet. This means that embedded hosts can control the configuration of a Digi device via the serial port. See pages 33 and 66.

Customization and integration scenarios

Chapter 4, "Scenarios" presents scenarios for customizing Digi devices and integrating the devices into your overall solution.

Customize Digi device look-and-feel

Scenarios for customizing the look-and-feel of your Digi device include:

- The web interface for Digi devices can be customized, for example, to display a different company logo, product name, or different background colors. The recommended process for changing the look of the web interface is to work with one device, customize the user interface, verify that the customized interface looks and operates as desired on the device, then upload the customized interface to other devices. See page 50
- Once the web interface is customized and verified in a single device, it can be uploaded to other devices. See page 51.
- For Digi devices that have a multiple-user model, there is a username and password defined specifically for performing customization tasks. See page 52.
- You can create a set of default settings for your Digi devices that differ from the factory defaults assigned by Digi, and load these custom factory defaults onto other devices. See page 53.

If the default web interface does not meet your needs, a sample Java-based configuration applet is available in the Digi Connect Integration Kit. You can customize this configuration applet and upload it to devices, and set the applet as the default user interface for the device. You can also use the sample configuration applet as a model for other applets developed for your devices. For more information on the applet, see pages 22 and 55

- If necessary, you can return a customized Digi device to its factory-default state. See page 58.

Manufacturability

Scenarios involving manufacturability include:

- "What do I need to set up a manufacturing environment?" on page 60.
- "How do I upload a customized web interface to other devices?" on page 59
- "How do I upload custom factory defaults to multiple devices?" on page 59
- "How do I upload an applet to multiple devices?" on page 59
- "How do I load a preferred firmware image during manufacturing?" on page 59

Implement software interfaces and redeploy elements

Scenarios for implementing software interfaces and redeploying elements include:

- "What software interfaces are available on Digi devices?" on page 61
- "How does device discovery work, and how can it be modified?" on page 61, including:
 - "What is Advanced Digi Discovery Protocol (ADDP)?" on page 61
 - "Which ports does ADDP use?" on page 61
 - "What are my options for interfacing with ADDP?" on page 62
 - "How can I modify the sample discovery program (ADDP)?" on page 62
 - "How can I discover and configure devices from VisualBasic® or other languages?" on page 62
- "What is the Remote Command Interface (RCI)?" on page 63
- "Which port does RCI use?" on page 63
- "Can I configure and monitor devices using SNMP?" on page 63
- "Can I redeploy RealPort drivers?" on page 63
- "How can I customize the product documentation?" on page 64

Implement hardware interfaces

While the focus of this Guide is more on the software-related tools available for customization and integration, it does cover several hardware-related topics:

- "How should I implement reset functions on my Digi device?" on page 65
- "How can I use RCI over serial?" on page 66
- "How can I customize a Digi Module?" on page 75 .

This chapter describes the tools and interfaces available for device customization and integration. The descriptions note where to find additional information on the tools and interfaces, such as other documents, web pages, and other Digi device product information.

Product look-and-feel customization tools

Several tools are available for customizing the look-and-feel for a Digi device, including

- Web interface customization tools, such as a web page for managing customization actions and files
- Custom factory defaults for device configuration settings
- The Java configuration applet, an alternate user interface that can be completely customized to your needs

Web interface customization tools

Several tools exist for customizing the web interface for devices, including:

- The **Customization Administration** page in the web interface. This page displays the history of customization actions for the device, and is used to download and upload the files involved in customization, such as the configuration file, custom factory defaults, custom stylesheet, and logo files.
- A Configuration File Template, named **config.ini**, for customizing identity-related content in the user interface, such as the product name, company name, and logo displayed on the device user interface.
- A Custom Factory Defaults Template, named **defaults.rci**, for customizing the default settings for the Digi device.
- A stylesheet template, which can be any filename but must have the extension **.css**.
- A file named **passwd.ini** that is created if the user name and password for logging onto the Customization Administration page is changed.

Customization Administration web page

On the **Customization Administration** page of the Digi device web interface, you can:

- Review a history of customization actions.
- Download template files for configuration, custom factory defaults, and stylesheets.
- Uploading the configuration, custom factory defaults, and stylesheets.



Access to this page depends on the user model for your Digi device.

Determining the user model for your Digi device

There are three user models implemented in the various Digi devices: one-user model, two-user model, and more than two-user model. To determine the user model, Telnet to the Digi device's command line and issue a **show user** or **set user** command. In the command output, note how many user IDs are defined: one, two, or more than two.

Access on single-user Digi devices

For Digi devices that support a single user accessing the device at a time, access the Customization Administration page at the following location:

`http://<ip address>/admin/customization.htm`

There is no special username and password for customization on single-user Digi devices. The default is no login prompt, and the default name for user 1 is **root**. Use of a login depends on whether user authentication is enabled a password is defined. These settings are in the web interface for the Digi device under **Configuration > Security**, in the Username, **Enable password authentication**, and Password settings.

Access on multiple-user Digi devices

For Digi devices that support multiple users, such as Digi Connect EM, Digi Connect ME, Digi Connect SP, etc., on the Home page of the Digi device's web interface, choose **Customization**. You are redirected to a login page for Customization Administration. The default username for the **custom** and the default password is **custom**.

Change customization username and password

For Digi devices that have a multiple-user model, there is a special username and password for customization tasks only. Changing this username and password from the default is recommended. See page 52. Changes are stored in a file called **passwd.ini**, which can be uploaded to other Digi devices using the Digi Connect Programmer's **/upload** command; see page 41.

Settings on the page

Customization Management shows a history of customization actions for the device. Any customization files that have been downloaded, modified, then uploaded back to the device, are displayed here.

The **Customization Settings** part of the page is used to download and upload files controlling the look-and-feel of Digi devices, including:

- The configuration file for the web interface, **config.ini**.
- The custom factory defaults file, **defaults.rci**.
- The custom stylesheet for the web interface, which can be any filename but must have the extension **.css**.
- If used, a custom logo to be displayed in the web interface.
- If you changed the customization username and password from the default, and wish to upload the changes to other Digi devices, file **passwd.ini** contains the changed user definition. See page 52 for more information.

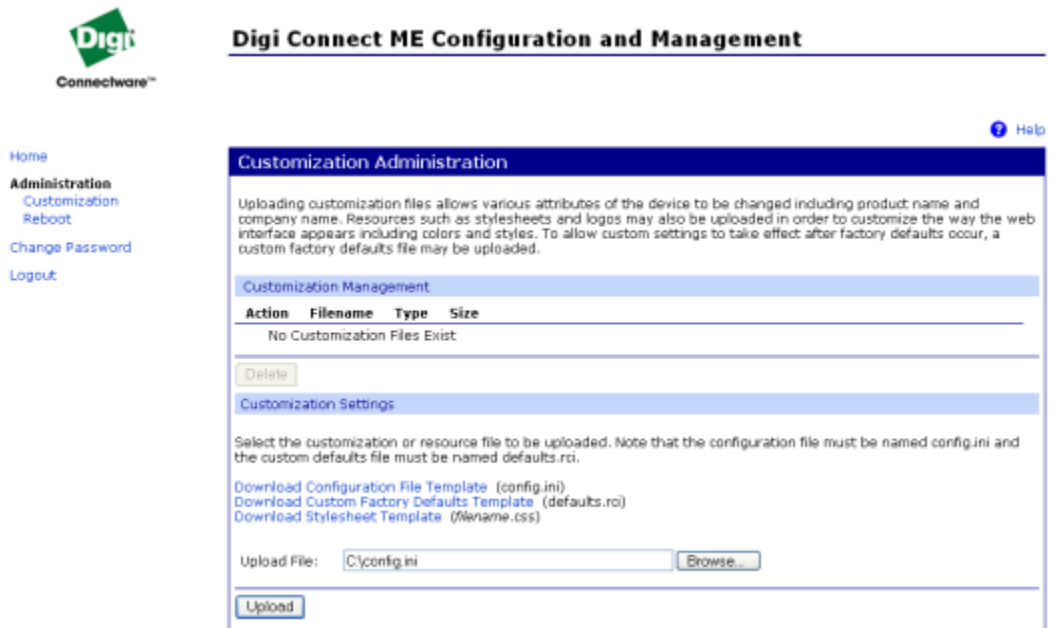
Download customization files

To download files for customization, select the desired file or files.

Upload customization files

To upload a file after customization actions, enter the filename or use the Browse button to locate it, then click **Upload**. The example shows a

config.ini file being uploaded to the Digi device.



Applet files uploaded from the File Management Page instead

Files for configuration applet or other Java-based applets are uploaded on the **File Management** page instead. These files can also be uploaded using the Digi Connect Programmer's **/upload** command and its **/dir** and **/file** parameters; see pages 24 and 41.

Configuration File Template (config.ini)

The Configuration File Template is a file with the required name **config.ini**. It defines the INI file structure and the Digi-specific defaults set in it, such as the company name, copyright information, product name, and links to documentation and other product resources.

You can use this file to define a custom logo, stylesheet, product names and company names to be displayed in the web interface for the device. This file allows the device to actually appear as a completely different device. You can then ship these units to customers without Digi being involved or without the user knowing the device is a Digi-OEM product.

Here is an example config.ini file. Most of the settings in this file are the Digi defaults, with two custom settings for the product name and product logo.

```
[Company]
LongName=Digi International Inc.
ShortName=Digi International
WebSiteURL=http://www.digi.com
WebSiteName=www.digi.com

Copyright=Copyright &copy; 1996-2005 Digi International Inc. All
rights reserved.
```



```
[Product]
ProductName=Digi Custom ME
ProductClass=Digi device server

[ProductLinks]
DocumentationURL=http://www.digi.com/support
FirmwareURL=http://www.digi.com/support
TechSupportURL=http://www.digi.com/support

[Resources]
ProductLogo=exclam.gif
ProductStylesheet=stylesheet.css

[Strings]
# Reserved for Future Use
```

How INI file settings work

An INI file is a collection of sections, each of which contains a group of *key=value* pairs. A section of an INI file defines a common group of settings. Those settings are then specified with a *key=value* pair. The *key* resembles the setting name and the *value* represents the name to assign to that *key*. Each section must be unique, and each *key* in a section must be unique as well. Thus, no two sections may share the same name, and no two keys in a section may share the same name. However, a key from a different section may match a key in another section, as they have completely different meanings.

For example, here are two sections from the config.ini file from the previous page:

```
[Company]
LongName=Digi International Inc.
ShortName=Digi International
WebSiteURL=http://www.digi.com
WebSiteName=www.digi.com
Copyright=Copyright &copy; 1996-2005 Digi International Inc. All
rights reserved.

[Product]
ProductName=Digi Custom ME
ProductClass=Digi device server
```

This example contains two sections: **[Company]** and **[Product]**. The **[Company]** section defines settings specific to the company developing the product, and the **[Product]** section defines settings specific to the product being developed. The **[Company]** section contains five *key=value* pairs, and the **[Product]** section, two. The settings show in this example are the Digi-assigned defaults. You can use these sections and settings to customize the company and product information displayed for your customized Digi device. Complete definitions for the config.ini sections and settings are on the following page.

Changing settings in the config.ini file

The easiest way to change configuration settings in the **config.ini** file is to perform the following steps:

1. From the Customization Administration page, download the **config.ini** file from the Digi device.
2. Using a text editor, edit the file. Settings and the Digi defaults are described on page 19. You can replace the Digi defaults with your own.
3. Save the changes to the file.
4. From the **Customization Administration** page, upload **config.ini** to the Digi device. Additionally, upload any custom stylesheets, images, logos, etc.
5. Reboot the device (required).

config.ini file settings


The table describes sections and *key=value* pairs in the Digi **config.ini** file. The file contains, at present, five sections:

- **Company:** Settings specific to the company developing the product.
- **Product:** Settings specific to the product being developed.
- **ProductLinks:** Various links to various product information.
- **Resources:** Resources such as stylesheets, logos, etc.
- **Strings:** Currently not being used and reserved for future use.

For all values, the following rules apply:

- When the item of a particular section is missing or omitted, the specified default value for the item is used.
- When the item of a particular section is specified, but the value is left empty, that item is not visible within the various interfaces.
- When the item and value of a particular section are specified, the specified value overrides the default value.

Section	Setting (key-value pair)	Description	Required	Default
Company	LongName	Long version of the company name used throughout product information.	Yes	Digi International Inc.
	ShortName	Short version of the company name, used in cases where a smaller string is needed for length purposes. LongName and ShortName can have the same value, but the ShortName value should not be more than 24 characters long. ShortName can also be an abbreviation for convenience.	No	Same as LongName.
	WebSiteURL	Full URL for the main company website.	Yes	http://www.digi.com
	WebSiteName	Display name for the web page address. This is the text used when a link to the company web page is used. Typically, the value is a simpler form of the URL. However, a descriptive name such as Company Website may be used.	Yes	www.digi.com
	Copyright	Copyright date for the company for the product.	Yes	Copyright © 1996-2005 Digi International, Inc. All rights reserved.

Section	Setting (key-value pair)	Description	Required	Default
Product	ProductName	Full name of the product.	Yes	Depends on the Digi product.
	ProductClass	Class of products. Often a company markets several products in a certain class. For example, Digi products often fall under Terminal Server or Device Server. This string defines that class.	Yes	Depends on the Digi product.
ProductLinks	DocumentationURL	Full URL for the website containing the product documentation. If no URL exists, a general URL to the company's technical support page may be used.	No	http://support.digi.com
	FirmwareURL	Full URL for the website containing firmware updates.	No	http://support.digi.com
	TechSupportURL	The full URL for the website containing technical support information for the product or the general technical support website for the company.	No	http://support.digi.com
Resources	ProductLogo	The filename of a custom logo that has been uploaded to the device. This string should contain filename only. All path or directory information should be removed. If the file has not been uploaded into the system, no image will be displayed to the user and there may be an unwanted effect on the user interface.	Yes	File logo.gif contains the Digi company logo, which looks like this: 
	ProductStylesheet	The filename of a custom stylesheet that has been uploaded to the device. This string should contain the filename only. All path or directory information should be removed. If the file has not been uploaded into the system, no image will be displayed to the user and there may be an unwanted effect on the user interface.	Yes	File stylesheet.css is the default Digi stylesheet.
Strings	None	Currently, no settings are available in this section. It will be used in future versions and defined at a later time. If desired, remove this section from configuration files.	No	None

Custom Factory Defaults Template file (defaults.rci)

The custom factory default file specifies the custom settings that are established when a user resets a Digi device to factory defaults, described below.

Stylesheet Template (.css)

Stylesheets define the look, layout, and colors of a web page. There is a stylesheet supplied in the factory default device version, called **stylesheet.css**

To customize this stylesheet, you need to edit it. Novice HTML developers will probably want to use an HTML-editor application, while more experienced users can directly edit the HTML code. Whoever edits the stylesheet should have a good understanding of HTML and the typical sections and commands in a stylesheet. If your command of HTML is limited, you might want to have an experienced Web developer handle this task.

For example stylesheets, download the Digi stylesheet from this URL:

`http://<ip address>/style/stylesheet.css`

Custom factory defaults

Custom factory defaults are customized default settings for Digi devices that differ from the Digi-assigned defaults. These are the settings to which the device reverts when it is reset to factory defaults. Custom factory defaults allow you to specify your own default settings so that when you deploy Digi devices to your customers, they will see your company's defaults, rather than the Digi-assigned defaults.

The custom factory defaults are stored in a file named **defaults.rci**. This file is a Remote Command Interface (RCI) file that defines the custom factory default settings to use for a device when it is powered up, and also when it is reset to factory defaults. From the Customization Administration page, you can download the defaults.rci file, and upload it to other devices as needed. Direct editing of this file is not recommended. Such editing requires a solid understanding of RCI. For more information about RCI, refer to the *Remote Command Interface (RCI) Specification*.

From the **Customization Administration** page, download file **defaults.rci** file, and upload it to other devices as needed.

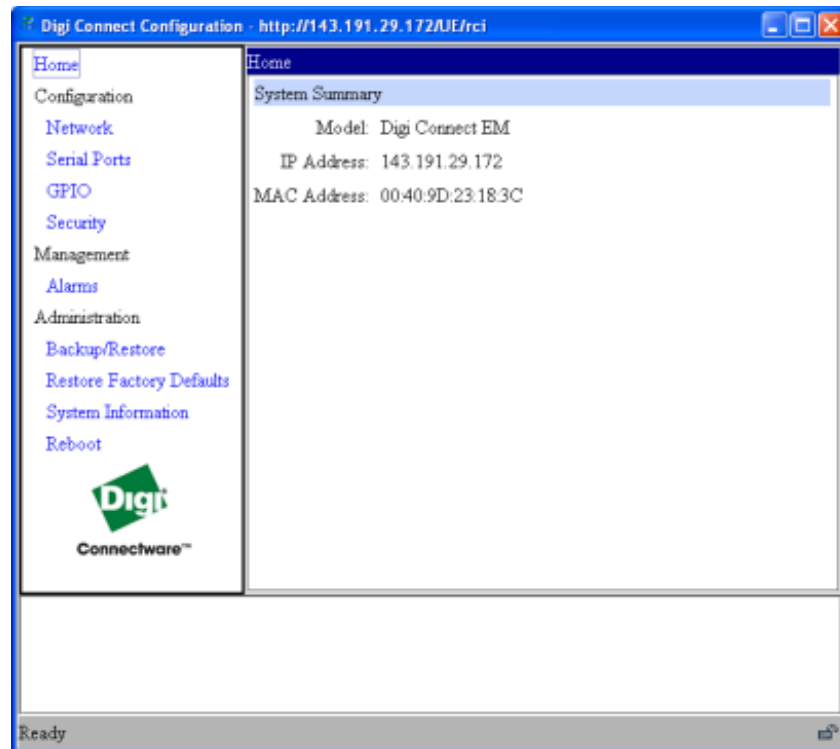
Exercise caution in changing device settings from the Digi-assigned defaults. Changing certain settings can render the device inaccessible. In addition, you should verify that the device operates appropriately with the changed defaults.

For more information on creating and using custom factory defaults, see page 53.

Configuration applet

The Digi Connect Integration Kit includes several sample applets. One of them is an alternative device configuration interface in the form of a Java applet. This interface provides many, but not all, of the configuration choices available through the default web interface.

The configuration applet is not available for Digi Cellular Family products. Here is the home page of the configuration applet.



To view a description of the configuration applet, go to **Start > Programs > Digi Connect > Samples > Java Applets > ConfigApp documentation**.

To view the applet source files, go to **Start > Programs > Digi Connect > Samples > Java Applets > ConfigApp documentation**.

The applet is a sample alternate device interface that can be used as a base for a custom user interface to create a unique user interface representing the kind of device in which the Digi device is embedded. For example, the configuration interface for a printer would look nothing like the Digi device's web interface. Today, the only way to create a totally custom interface to the device is through an applet or a Remote Command Interface (RCI) application. The applet can be slightly modified using a configuration file, or it can be changed extensively. In addition, it can be used as a sample by those customers who choose to write their own custom configuration.

The configuration applet does not include all features and settings of the web interface and applet code is provided as-is. There is no written or expressed warranty or support for this code. You can examine and extend the code as desired, but note that future releases of this code may be changed without notice.

How the configuration applet works

The configuration applet is built on a set of Digi utility classes in the `common.jar` library. The classes available in this library are described in the accompanying java documentation. Code can be built and run as an applet or as a standalone application.

The configuration applet runs by pointing a browser at the IP address of the device server. This opens the **index.htm** file on the device which in turn downloads and invokes the applet within the browser's Java virtual machine (JVM) plug-in. This applet is written such that it requires the Sun Java plug-in version 1.4.2 or newer. This was done to take advantage of the advanced UI and networking features available in this JVM. Your custom written applets and applications can be written to run on JVMs offered by other vendors as you see fit.

The configuration applet was developed with a number of extension points in mind. The current implementation allows customizing some of the look-and-feel of the user interface without having to re-code the applet. Changing string labels, changing screen colors, or setting background images can all be done easily without changing a line of code.

Configuration applet-related files

Several files are involved with the configuration applet interface:

File	Description
common.jar	A shared library that provides the underlying communication and device management classes. For details about jar file organization and using the classes, go to these javadoc links: configapp.jar javadoc common.jar javadoc
configapp.jar	File in which the configuration applet is packaged.
index.htm	File opened when the configuration applet is started.
config.ini	Configuration file for the applet, defining the look-and-feel of the applet interface, such as the background color. This is a different file than the one of the same name used for customizing the web interface.
Resource .properties files	Define labels and message text in the configuration applet's user interface.
.java source files	Define pages and content of the configuration applet.
.default	Hidden file created if the configuration applet is set as the default user interface.

Customizing the configuration applet

There are several ways to customize the configuration applet:

- To customize the look-and-feel, edit file **config.ini**.
- To customize the labels and message text in the user interface, edit the resource **.properties** files.
- To create custom views and edit content, edit the **.java** source files.

Uploading applet files to Digi devices

The applet files need to be uploaded into Digi devices, either through the **File Management** page of the web interface, or the Digi Connect Programmer **/upload** command. Using **/upload** command is more efficient when uploading the applet files to many Digi devices.

Setting the configuration applet as the default user interface

The configuration applet can be set as the default user interface for a device. On the Home page of the default web interface, there is a section titled **User Interfaces**. On the line for Custom Interface, click the **Set as Default** button. This action creates a file named **.default**. This is a hidden file.

Manufacturability tools and resources

The Digi Connect Integration Kit includes several tools for efficiently configuring and deploying devices in the manufacturing process, including:

- The capability to load a preferred firmware image into Digi devices
- Digi Connect Programmer, a command-line interface for efficiently loading custom factory defaults and other customization files to Digi devices
- A sample manufacturing environment for mass manufacturing and device configuration

Load preferred firmware image

A preferred firmware image is a version of Digi device firmware that a customer prefers to have loaded in their Digi devices. This could be a firmware version level that is several versions behind the currently shipping version. Your company takes responsibility for loading this firmware into devices, via the Digi Connect Programmer, described on the following page. For more information, see “How do I load a preferred firmware image during manufacturing?” on page 66.

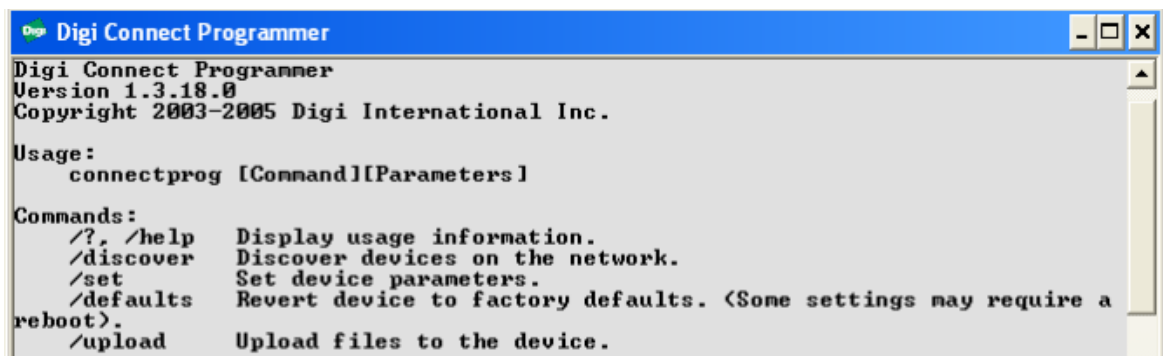
Digi Connect Programmer

Digi Connect Programmer is a command-line-based interface to Digi devices included in the Digi Connect Integration Kit. It runs on Microsoft® Windows® only. Chapter 3, "Digi Connect Programmer" describes its commands and parameters. Digi Connect Programmer commands can be used to:

- Discover devices on the network. The Digi Connect Programmer uses the Advanced Digi Discovery Protocol (ADDP) to discover devices.
- Set network parameters for the device, for example, to set a temporary IP address, subnet mask, and default gateway for the Digi device for loading customization files.
- Upload files to devices, such as customization files, custom factory defaults, Java applet files, and firmware images. From a customization perspective, the main use of the Digi Connect Programmer is to efficiently manage uploading customization files and firmware images to one or more Digi devices. After files have been downloaded, changed, and uploaded to a Digi device, Digi Connect Programmer can upload these customization files to other Digi devices, helpful when loading files onto many devices.
- Revert device settings to factory defaults.
- Reboot the device.
- Execute commands in scripts that perform device-manufacturing operations in a programmed sequence. The sample manufacturing environment included with the Digi Connect Integration Kit demonstrates such scripts.

To access the Digi Connect Programmer, select

Start > Programs > Digi Connect > Digi Connect Programmer. A command window opens, displaying help for the program:



```

Digi Connect Programmer
Version 1.3.18.0
Copyright 2003-2005 Digi International Inc.

Usage:
    connectprog [Command][Parameters]

Commands:
    /?, /help      Display usage information.
    /discover      Discover devices on the network.
    /set           Set device parameters.
    /defaults      Revert device to factory defaults. (Some settings may require a
reboot).
    /upload        Upload files to the device.
  
```

Commands

Chapter 3, "Digi Connect Programmer" describes Digi Connect Programmer interface and its commands; in summary, the commands are:

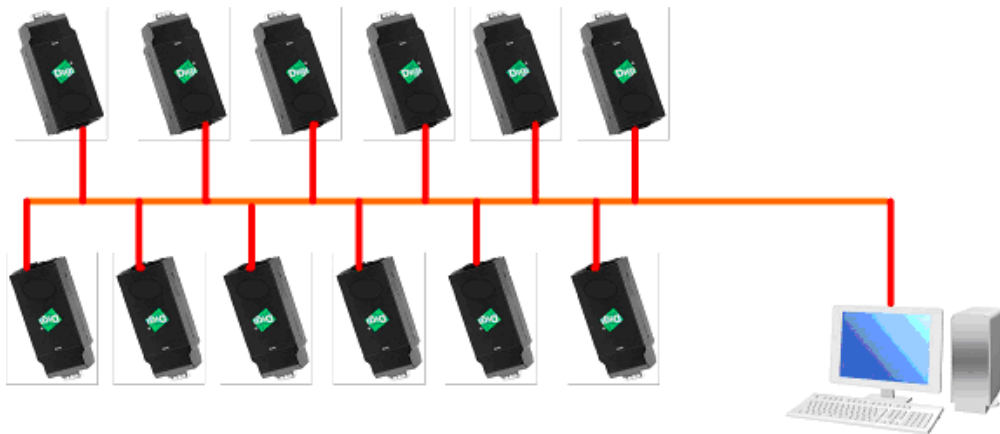
- **/discover**: Discovers Digi devices on a network; see page 39.
- **/set**: Sets network parameters (IP address, subnet mask, default gateway) in the Digi device; see page 40.
- **/upload**: Uploads files to the Digi device, including customization files, custom factory defaults, applet files, and firmware images; see page 41.
- **/defaults**: Reverts the Digi device's settings to factory defaults; see page 45.
- **/reboot**: Reboots the Digi device; see page 46.
- **/?** and **/help**: display help text.

Sample manufacturing environment

Once the Digi device is customized, the next task is to efficiently configure and deploy this custom user interface on other Digi devices. The Digi Connect Integration Kit includes a sample manufacturing environment. This sample demonstrates the components involved in configuring and deploying multiple Digi devices. You can build upon and modify this sample to set up your own manufacturing environment. To access the sample manufacturing environment, select **Start > Programs > Digi Connect > Samples > Manufacturing Tools**.

Sample manufacturing environment components

The sample manufacturing environment represents the following equipment setup:



Components of this manufacturing environment include:

- A manufacturing PC that runs applications and scripts for configuring devices.
- Digi devices to be configured with custom user interface software and custom factory defaults.
- A private network, with Dynamic Host Configuration Protocol (DHCP) enabled for IP address assignment. For wireless devices, a DHCP server for assigning IP addresses is particularly useful. A DHCP server is available at the Web site for Tftp32 at this location:

<http://perso.wanadoo.fr/philippe.jounin/tftpd32.html>

- The Digi Connect Programmer
- A set of scripts running on the manufacturing PC. The scripts perform a variety of operations, such as displaying prompts for user input on the manufacturing PC, using ADDP to discover devices, and executing Digi Connect Programmer commands to load customization files. A readme file accompanies the scripts and describes how to use them. The sequence of actions performed by the scripts is:
 1. Use ADDP to discover the devices on the network.
 2. Load custom user interface files and update firmware.
 3. Reboot the device.
 4. Log success or failure of the process.
 5. Repeat this process as necessary for one or more devices.

The same procedure can be used to assign IP addresses to devices.

Modifying sample manufacturing environment code

You can modify and extend the scripts and code in the sample manufacturing environment to suit your company's needs. The scripts include comment text to help you understand and modify the code.

Software interfaces and redeployable elements

Digi devices support several common interfaces to devices, such as Telnet, raw TCP sockets, UDP sockets, RealPort, HTML, FTP, SSL, and SNMP. These common interfaces are described in *User's Guide* for the Digi device. In addition to these common interfaces, there are several Digi-proprietary interfaces:

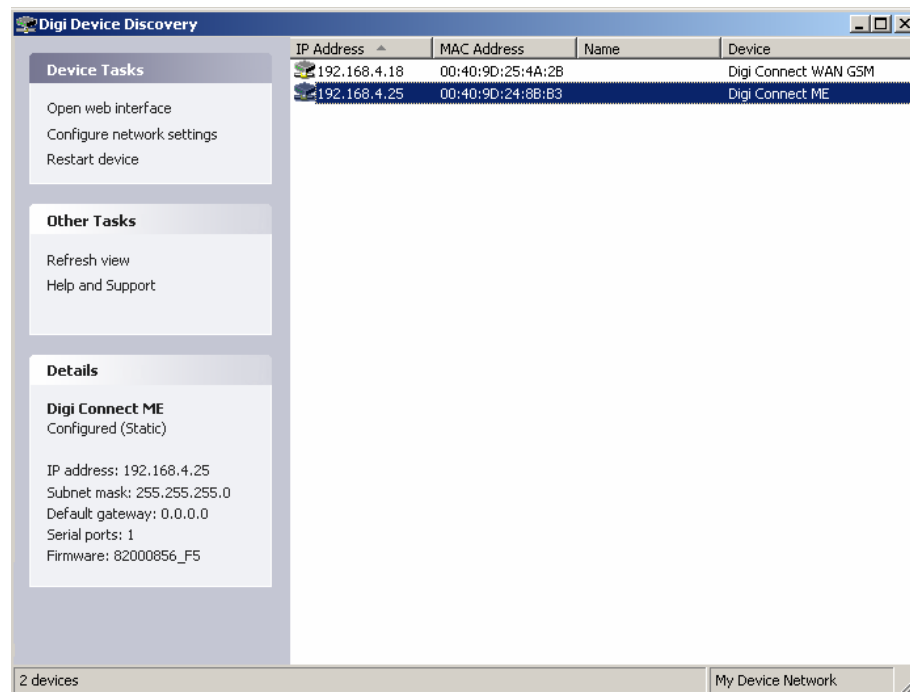
- Advanced Digi Discovery Protocol (ADDP), a Digi International-proprietary protocol for discovering devices on networks.
- Remote Command Interface (RCI), a programmatic interface for configuring and controlling Digi devices

Device configuration and monitoring through Simple Network Management Protocol (SNMP) involves both standard Management Information Blocks (MIBs) and several Digi-specific MIBs.

Also, there are elements of the standard Digi devices that can be redeployed in your company's software distribution, including RealPort drivers and Digi device product documentation.

Advanced Digi Discovery Protocol (ADDP) - Device discovery program

ADDP stands for Advanced Digi Discovery Protocol. ADDP is a Digi International-proprietary protocol for discovering devices on networks. ADDP runs on operating system capable of sending multicast IP packets on a network. It allows the system to identify all ADDP-enabled Digi devices attached to a network by sending out a multicast packet. The Digi devices respond to the multicast packet and identify themselves to the client sending the multicast.



Options for interfacing with ADDP

ADDP functions are supplied in two formats, allowing for flexibility in implementing the functions in applications:

- As static library (**.lib**) files. The ADDP functions are directly supplied in your application files, and you do not have to ship extra files. The drawback to using these files is that there is limited flexibility. The functions can only be linked to Microsoft Visual C++[®] applications.
- As Dynamically Linked Library (**.dll**) files associated with the application. Using these files allows for more flexibility, as the functions can be linked to interfaces other than Microsoft Visual C++ applications. However, this approach also means more files must be associated with the main application at runtime.

Additional ADDP information

For more information on ADDP, see page 62, and the “C” header file for ADDP in the Digi Connect Integration Kit Samples (**Start > Programs > Digi Connect > Samples > Discovery > CPP > Device Discovery Documentation**).

Remote Command Interface (RCI)

Remote Command Interface (RCI) is a programmatic interface for configuring and controlling Digi devices. RCI is an XML-based request/response protocol allowing a caller to query and modify device configurations, access statistics, reboot the device, and reset the device to factory defaults.

Unlike other configuration interfaces that are designed for a user, such as the command-line and web interfaces, RCI is designed to be used by a program. Examples RCI uses include calls within a Java applet stored on the Digi device that provides a custom configuration interface to the device, a custom application running on a PC that monitors and controls installation of many Digi devices, or utilities to configure or initialize devices through external programs or scripts.

RCI uses HTTP as the underlying transport protocol.

Configuration requests are made in Extensible Markup Language (XML) format.

Not all actions taken through the default web interface have direct equivalents in RCI.

Several RCI resources are available:

- The *Remote Command Interface (RCI) Specification* lists all of the settings and states accessible through RCI.
- The source code files for the configuration applet, available from the Samples menu for the Digi Connect Integration Kit, provides an example of using RCI in application/applet code.

Simple Network Management Protocol (SNMP)

Simple Network Management Protocol (SNMP) is a protocol for managing and monitoring network devices. The SNMP architecture enables a network administrator to manage nodes--servers, workstations, routers, switches, hubs, etc.--on an IP network; manage network performance, find and solve network problems, and plan for network growth. Digi devices support SNMP Version 1.

Advantages and disadvantages of using SNMP

The advantages of using SNMP as a device configuration and management interface include:

- SNMP is easy to implement in extensive networks.
- Programming new variables is easy.
- SNMP is widely used. SNMP is a standard interface that integrates well with network management stations in an enterprise environment. While its capabilities are limited to device monitoring and display of statistics in Digi devices, read/write capabilities are expected to be added to Digi devices in future releases.
- It is easy to 'drop in' new devices.

Some of the drawbacks to using SNMP as a device configuration and management interface include:

- As device communication is UDP-based, the communication is not secure. If you require more secure communications with a device, you will need to use an alternate interface.
- Using SNMP, you cannot do as many tasks as you can from the default web interface, such as file management, uploading firmware, or backing up/restoring configurations.
- Compared to the default web interface or the command-line interface, SNMP is limited in its ability to set specific parameters, such as set port profile, is not possible.

Access the SNMP Interface

Accessing the SNMP interface requires a tool, such as a network management station. The management station relies on an agent at a device to retrieve or update the information at the device, including Device configuration, status, and statistical information. This information is viewed as a logical database, called a Management Information Base (MIB). MIB modules describe MIB variables for a variety of device types and computer hardware and software components.

Standard MIBs supported

The standard MIBs supported in Digi devices are:

- **MIB-II (RFC 1213)** This is a MIB for managing a TCP/IP network. It is an update of the original MIB, now called MIB-I. MIB-II contains variable definitions that describe the most basic information needed to manage a TCP/IP network. These variable definitions are organized into several groups, such as groups for managing the system, network interfaces, address translation, transmission media, and various protocols, including IP, ICMP, TCP, UDP, EGP, and SNMP.
- **CHARACTER-MIB** (RFC 1658).
- **RS-232-MIB** (RFC 1659).

Digi enterprise MIBs supported

In addition to the standard MIBs, Digi devices use several Digi enterprise MIBs, including:

- **DIGI-DEVICE-INFO.mib** for handling device information. This MIB gives access to elements like the firmware revision, device name, IP network information, memory, and CPU statistics.
- **DIGI-SERIAL-ALARM-TRAPS.mib** for handling alarms sent as SNMP traps.

Device monitoring through SNMP

Device monitoring capabilities from SNMP include, among other things:

- Network statistics, defined in RFC 1213, MIB-II.
- Port statistics, defined in RFCs 1316 and 1317.
- Device information, defined in Digi enterprise MIB **DIGI-DEVICE-INFO.mib**.

For more information on the statistics available through the standard RFCs listed above, refer to the RFCs available on the IETF web site, **www.ietf.org**. For enterprise MIBs, see the description fields in the MIB text.

Additional SNMP information

A variety of resources about SNMP are available, including reference books, and overviews and other files on the Internet. For an overview of the SNMP interface and MIB-II components, go to **<http://www.rfc-editor.org/rfcsearch.html>**, and enter **MIB-II**. From the results, navigate to the text file describing the SNMP interface, titled **Management Information Base for Network Management of TCP/IP-based internets: MIB-II**. You can also view the text of the Digi enterprise MIBs.

RealPort drivers

Digi devices use the patented RealPort COM/TTY port redirection for Microsoft Windows, UNIX, and Linux environments. RealPort software provides a virtual connection to serial devices, no matter where they reside on the network. The software is installed directly on the host and allows applications to talk to devices across a network as though the devices were directly attached to the host. Actually, the devices are connected to a Digi device server or terminal server somewhere on the network.

RealPort is unique among COM port re-directors because it is the only implementation that allows multiple connections to multiple ports over a single TCP/IP connection. Other implementations require a separate TCP/IP connection for each serial port. Unique features also include full hardware and software flow control, as well as tunable latency and throughput.

Access to RealPort services can be enabled or disabled.

The Digi RealPort drivers used in Digi devices are available for redeployment in your own company's software distribution. See "Can I redeploy RealPort drivers?" on page 63.

Digi device product documentation

The standard product documentation for Digi devices includes:

- *Remote Command Interface (RCI) Specification*: describes the data elements of the Remote Command Interface protocol that can be used as a programmatic interface to Digi devices.
- *User's Guide* for the Digi device family: describes the standard features, functions, and interfaces.
- *Digi Connect Family Command Reference*: describes the standard command-line interface; applies to all device families.
- *Hardware Reference Manual* manuals for Digi devices. Several of the Digi devices have manuals that describe the hardware characteristics the products, and the capabilities and programming considerations of these modules in integration efforts.

If you need to make significant changes to the user documentation to reflect your customization and integration efforts, Digi International can provide you with the source for these documents.

Hardware interfaces

Several hardware-related features are available for use in customization and integration efforts.

Reset functions

Hard and soft reset functions can be implemented for Digi devices. For details, see page 65.

RCI over serial

The intention behind RCI over serial is to provide a mechanism for embedded hosts to control the configuration of a Digi device via the only data interconnect between the two; namely, the serial port. One of the serial pins is allowed to become a control signal, telling us that data being received on the serial port should be interpreted as RCI queries. The standard XML formatted RCI responses are returned to the embedded host via the serial port for parsing, if necessary. For more information, see page 66.

Digi Connect Programmer is a command-line interface to Digi devices included in the Digi Connect Integration Kit. It runs on Microsoft Windows only. Digi Connect Programmer commands to:

- Discover devices on the network. The Digi Connect Programmer uses the Advanced Digi Discovery Protocol (ADDP) to discover devices.
- Set network parameters for the device, for example, to set a temporary IP address, subnet mask, and default gateway for the Digi device for loading customization files.
- Upload files to devices, such as customization files, custom factory defaults, Python™ files, Java applet files, and firmware images.
- Revert device settings to factory defaults.
- Reboot the device.
- Use Digi Connect Programmer commands in scripts that perform device-manufacturing operations in a programmed sequence. The sample manufacturing environment included with the Digi Connect Integration Kit demonstrates such scripts.

Start Digi Connect Programmer

To start Digi Connect Programmer from the Start menu, select **Start > Programs > Digi Connect > Digi Connect Programmer**. A command window opens, displaying brief help for the program:



```

Digi Connect Programmer
Version 1.3.18.0
Copyright 2003-2005 Digi International Inc.

Usage:
    connectprog [Command][Parameters]

Commands:
    /?, /help      Display usage information.
    /discover      Discover devices on the network.
    /set           Set device parameters.
    /defaults      Revert device to factory defaults. (Some settings may require a
reboot).
    /upload        Upload files to the device.

Parameters for selection of target device:
    /mac           MAC address of the Digi Connect device.
                   Example: /mac=auto (program first device found)
                   Example: /mac=00409D123456 (program individual device)
    /destip        Current IP address of the Digi Connect device.
                   Example: /destip=192.168.1.2
    /vendorid      Digi assigned vendor identifier.
                   Example: /vendorid=0x44494749 (Digi default)

Parameters for authentication:
    /username      Username to login to the device (default: <none>).
                   Example: /username=root
    /password      Password to login to the device (default: <none>).
                   Example: /password=dbps

Parameters for configuring network settings:
    /ip            Temporary IP address of device during manufacturing.
                   Example: /ip=192.168.1.2
    /subnet        Temporary subnet mask of device during manufacturing.
                   Example: /subnet=255.255.255.0
    /gateway       Temporary gateway address of device during manufacturing.
                   Example: /gateway=192.168.1.1

Parameters for file handling:
    /dir           Directory name of files to upload to the device.
                   Example: /dir=c:\mydir
    /file          File name of file to upload to the device.
                   Example: /file=c:\myfile.txt
    /config        File name of configuration to upload to the device.
                   Example: /config=c:\myconfig.rci
    /firmware      File name of firmware (EOS) to flash the device with.
                   Example: /firmware=c:\myfirmware.bin
    /custom        File name of customization file to upload to the device.
                   Example: /custom=c:\stylesheet.css
    /erasefs       Erases the file system prior to uploading new files.
    /erasecustom   Erases the custom files prior to uploading new files.

    /silent        Allow silent configuration (no text output).
    /pause         Wait for key press after manufacturing cycle.
    /reboot        Reboot device.
  
```

General usage information

- The Digi Connect Programmer tool has been renamed **connectprog.exe**. It was formerly named **dgdcp prog.exe**. If you have any legacy scripts with references to **dgdcp prog**, update them with the new name.
- Digi Connect Programmer runs on Microsoft Windows only.
- All commands must be preceded by “connectprog” and all commands and parameters are preceded by a / character. For example:

```
connectprog /discover /vendorid=0x44494749
```
- On commands where device identification is required, devices can be identified either by their IP address or MAC address. This choice is represented in the command syntax as:

```
{/destip=ip address | /mac=mac address}
```
- Optional parameters are enclosed in brackets.
- Several commands have **/username** and **/password** parameters. Use of these parameters depends on whether the Digi device supports multiple users and has user authentication enabled.
- Several parameters can be used as flags in the commands, for example, to suppress output or pausing for a user to press a key after completing the manufacturing process for a device when they are ready to move on to another device. These parameters are not shown on the syntax statements for the commands, but are covered separately in "Parameters used as flags" on page 38.
- For descriptions of error messages and how to resolve them, see page 48.
- **Device configuration is limited through this interface.** Digi Connect Programmer allows setting the IP address, subnet mask, and default gateway for a Digi device. Any other device configuration settings must be configured through other device interfaces, such as web or command-line interface. For example, if you need to enable or disable DHCP for a particular device, you must do that through other means, such as the web interface's Network Configuration page or the **set network** command, through scripting, or silently as part of the installation process.

Digi Connect Programmer parameters

Digi Connect Programmer command descriptions identify required and optional parameters. Several parameters can be specified on most commands, and several parameters serve as flags to perform actions in manufacturing scripts.

For specifying the target device

- ***/destip=ip address*** or ***/mac={auto|mac address}***

Either */destip* or */mac* must be specified on all commands except */discover* to identify the destination Connect device.

- ***/vendorid=vendor id***
- ***/username=username***
- ***/password=password***

Use of */username* and */password* parameters depends on whether the Digi device supports multiple users and whether user authentication is enabled.

For configuring network settings

- ***/ip=ip-address***
- ***/subnet=subnet mask***
- ***/gateway=gateway address***

For uploading Web customization and configuration files

- ***/config=filename.rci***
- ***/custom=filename.rci***
- ***/erasecustom***

For uploading file systems

- ***/dir=directory name***
- ***/file=filename***
- ***/erasefs***

For uploading firmware files

- ***/firmware=filename.bin***

Parameters used as flags

These parameters can be used as flags in the programming process to perform actions, such as waiting for user input after an iteration of a manufacturing process completes.

/erasefs

Erases the file system in the Digi device before uploading new files.

/erasecustom

Erases all customization and configuration files in the Digi device, including config.ini, defaults.rci, the custom logo file, custom stylesheet (.css) files, and the passwd.ini file.

/pause

Wait for key press after manufacturing cycle. When used with the **/mac=auto** parameter, this flag allows the user performing device-manufacturing operations to replace the current device and begin the manufacturing process with a new device.

/reboot

Reboots the Digi device after manufacturing; performs the same operation as the **/reboot** command.

/silent

Allow silent configuration (no text output). This flag is useful if the **connectprog** command is invoked programmatically from another application.

/discover - Discover devices

Purpose	<p>Discovers Digi devices on a network.</p> <p>The Advanced Digi Discovery Protocol (ADDP) is used to perform device discovery. Note that the results of device discovery are impacted by firewalls in place on the network involved. For example, the Microsoft® Windows® firewall may block responses from misconfigured devices.</p>
Syntax	<pre>connectprog /discover [/vendorid=<i>vendor id</i>]</pre>
Parameters	<p>/vendorid=<i>vendor id</i></p> <p>The default Digi-supplied vendor ID is 0x44494749. This value cannot be changed in the field.</p> <p>Custom vendor IDs can be assigned by Digi and manufactured into specific firmware builds. A custom vendor ID allows a vendor to search for and find only its products instead of all Digi and Digi-OEM products. Contact Digi Sales to get a custom vendor ID.</p>
Example	<p>Discover devices with the default Digi vendor ID on a network</p> <pre>connectprog /discover</pre>
See also	<ul style="list-style-type: none">• "Advanced Digi Discovery Protocol (ADDP) - Device discovery program" on page 28.• "Parameters used as flags" on page 38.• "Digi Connect Programmer error messages" on page 48.

/set - Set network parameters

Purpose Sets network parameters for the Digi device, including the IP address, subnet mask, and default gateway.

Syntax

```
connectprog /set {/destip=ip address|/mac=mac address}
    [/ip=ip address]
    [/subnet=subnet]
    [/gateway=gateway]
    [/vendorid=vendor id]
    [/username=username]
    [/password=password]
```

Parameters **/destip=*ip address***
The IP address of the Digi device. Either this address or the MAC address is required to identify the device.

/mac={*auto*|*mac address*}
The MAC address of the Digi device. Either this address or the IP address is required to identify the device.

auto

Program the first device found.

mac-address

Program a device with a specific MAC address, for example, /mac=00409D123456.

/ip=*ip address*

Temporary IP address of the device during manufacturing.

/subnet=*subnet mask*

Temporary subnet mask of the device during manufacturing.

/gateway=*gateway address*

Temporary gateway address of the device during manufacturing.

/vendorid=*vendor id*

Digi-assigned vendor identifier. See "/vendorid=vendor id" on page 39.

/username=*username*

Username for logging in to the device. Use depends on whether user authentication is enabled for the Digi device. Default is <none>.

/password=*password*

The password for logging in to the device. Use depends on whether user authentication is enabled for the Digi device. Default is <none>.

Example

```
connectprog /set /mac=auto /ip=192.168.1.2 /subnet=255.255.255.0
/gateway=192.168.1.1
```

For devices that require authentication, the "/username" and "/password" parameters must also be specified.

See also

- "Parameters used as flags" on page 38.
- "Digi Connect Programmer error messages" on page 48.

/upload - Upload files to devices

Purpose

Uploads files to Digi devices. Use to upload such files as:

- Web interface configuration and customization files
- Custom factory defaults, plus any additional .rci files containing settings to override the custom factory defaults.
- A backup copy of a device's current configuration settings (device settings when a backup copy is made through the web interface **Backup/Restore** operation, not factory or custom-factory defaults).
- Files or file systems, such as applet and application files
- Firmware images.
- Python files

A single **/upload** command can be used to upload several files, but note that **username** and **password** values for customization files differ from other file types.

Syntax

Upload Web customization and configuration files

```
connectprog /upload {/destip=ip address|/mac=mac address}
    [/config=filename]
    [/erasecustom]
    [/vendorid=vendor id]
    [/username=username]
    [/password=password]
```

Upload custom factory defaults

```
connectprog /upload {/destip=ip address|/mac=mac address}
    [/custom=filename.rci]
    [/vendorid=vendor id]
    [/username=username]
    [/password=password]
```

Upload a backup copy of current configuration settings

```
connectprog /upload {/destip=ip address|/mac=mac address}
    /config=filename
    /reboot
```

Upload file-system files

```
connectprog /upload {/destip=ip address|/mac=mac address}
    [/dir=directory name]
    [/file=filename]
    [/erasefs]
    [/vendorid=vendor id]
    [/username=username]
    [/password=password]
```

Upload firmware images

```
connectprog /upload {/destip=ip address|/mac=mac address}
                /firmware=filename
                [/vendorid=vendor id]
                [/username=username]
                [/password=password]
```

Upload Python files**Parameters****/destip=*ip address***

IP address of the Digi device. Either this address or the MAC address is required to identify the device.

/mac={*auto*|*mac address*}

MAC address of the Digi device. Either this address or the IP address is required to identify the device.

auto

Programs the first device found.

mac address

Programs a device with a specific MAC address, for example, /mac=00409D123456.

/config=*filename*

Name of an RCI-based file containing device configuration settings to upload. The file loaded by this parameter overrides settings established by the custom factory defaults file, defaults.rci, which would be uploaded through the **/custom** parameter. For example,

```
/config=c:\myconfig.rci.
```

Performs the same operation as the Restore operation in the web interface.

/custom=*filename.rci*

Specifies the name of an RCI-based file containing custom factory defaults to be uploaded to the device.

If the Digi device supports multiple users, you need to also supply the **/username** and **/password** parameters, specifying the username and password for performing customization activities and accessing the **Customization Administration** web page. For example:

```
/custom=defaults.rci /username=custom /password=custom
```

/erasecustom

Erases all customization and configuration files in the device, including **config.ini**, **defaults.rci**, the custom logo file, custom stylesheet (**.css**) files, and the **passwd.ini** file.

/dir=*directory name*

Directory name containing files to upload, for example:

```
/dir=c:\mydir
```

/file=*filename*

File to be uploaded. Use this parameter to upload files such as the files for a customized Java configuration applet. For example:

```
/file=c:\myfile.txt
```

/erasefs

Erases the file system in the device before uploading new files.

/firmware=filename.bin

Firmware file (EOS) to flash the device with. For example:

```
/firmware=c:\myfirmware.bin
```

Performs the same operation as the **Update Firmware** operation in the web interface.

/vendorid=vendor id

Digi-assigned vendor identifier; see page 39.

/username=username

Username for logging in to the device. Use depends on whether user authentication is enabled. The default is <none>.

/password=password

Password for logging in to the device. Use depends on whether user authentication is enabled. The default is <none>.

/reboot

Reboot device after loading the file containing configuration settings.

Examples**Upload Web customization and configuration files**

```
connectprog /upload /destip=192.168.1.1
/custom=c:\customization\stylesheet.css

connectprog /upload /destip=192.168.1.1
/custom=c:\customization\mylogo.gif

connectprog /upload /destip=192.168.1.1 /custom=c:\config.ini
```

Upload custom factory defaults

```
connectprog /upload /destip=192.168.1.1
/custom=c:\customization\defaults.rci /password=custom
/username=custom
```

Upload backup copy of current configuration settings

```
connectprog /upload /mac=auto /config=backup.rci /reboot
```

Upload file-system files

```
connectprog /upload /mac=00:40:9D:24:8B:B3
/file=c:/applet/config.ini
```

Upload firmware images

```
connectprog /upload /mac=auto /firmware=820001253.bin /reboot
```

See also

- "Configuration File Template (config.ini)" on page 16.
- "Custom Factory Defaults Template file (defaults.rci)" on page 21.
- "Stylesheet Template (.css)" on page 21.
- "Configuration applet-related files" on page 23.

- "Load preferred firmware image" on page 24.
- "Parameters used as flags" on page 38.
- "Digi Connect Programmer error messages" on page 48.

/defaults - Revert device to factory defaults

Purpose	Reverts device to factory defaults, or custom factory defaults if loaded in the device. Some settings may require rebooting the device as well.
Syntax	<pre>connectprog /defaults {/destip=<i>ip address</i> /mac=<i>mac address</i>} [/vendorid=<i>vendor id</i>] [/username=<i>username</i>] [/password=<i>password</i>] [/ip=<i>ip address</i>] [/subnet=<i>subnet mask</i>] [/gateway=<i>gateway address</i>]</pre>
Parameters	<p>/destip=<i>ip address</i> IP address of device. Either this address or the MAC address is required to identify the device.</p> <p>/mac={<i>auto</i> <i>mac address</i>} MAC address of device. Either this address or the IP address is required to identify the device.</p> <p>auto Program the first device found.</p> <p><i>mac-address</i> Program a device with a specific MAC address, for example, /mac=00409D123456.</p> <p>/vendorid=<i>vendor id</i> The Digi-assigned vendor identifier; see page 39.</p> <p>/username=<i>username</i> Username for logging in to the device. Use depends on whether user authentication is enabled. Default is <none>.</p> <p>/password=<i>password</i> The password for logging in to the device. Use depends on whether user authentication is enabled. Default is <none>.</p> <p>/ip=<i>ip address</i> Temporary IP address of the device during manufacturing.</p> <p>/subnet=<i>subnet mask</i> Temporary subnet mask of the device during manufacturing.</p> <p>/gateway=<i>gateway address</i> Temporary gateway address of the device during manufacturing.</p>
Example	<pre>connectprog /defaults /mac=00:40:9D:24:8B:B3</pre>
See also	<ul style="list-style-type: none">• "/reboot - Reboot device" on page 46.• To upload custom factory defaults, see pages 21 and 41.• "Parameters used as flags" on page 38.• "Digi Connect Programmer error messages" on page 48.

/reboot - Reboot device

Purpose	Reboots the device. Performs the same operation as /reboot parameter.
Syntax	<pre>connectprog /reboot {/destip=<i>ip address</i> /mac=<i>mac address</i>} [/vendorid=<i>vendor id</i>] [/username=<i>username</i>] [/password=<i>password</i>] [/ip=<i>ip address</i>] [/subnet=<i>subnet</i>] [/gateway=<i>gateway</i>]</pre>
Parameters	<p>/destip=<i>ip address</i> IP address of the device. Either this address or the MAC address is required to identify the device.</p> <p>/mac={<i>auto</i> <i>mac address</i>} MAC address of the Digi device. Either this address or the IP address is required to identify the device.</p> <p>auto Program the first device found.</p> <p><i>mac-address</i> Program a device with a specific MAC address, for example, /mac=00409D123456.</p> <p>/vendorid=<i>vendor id</i> Digi-assigned vendor identifier; see page 39.</p> <p>/username=<i>username</i> Username for logging in to the device. Use depends on whether user authentication is enabled. Default is <none>.</p> <p>/password=<i>password</i> Password for logging in to the device. Use depends on whether user authentication is enabled. Default is <none>.</p> <p>/ip=<i>ip address</i> Temporary IP address of the device during manufacturing.</p> <p>/subnet=<i>subnet mask</i> Temporary subnet mask of the device during manufacturing.</p> <p>/gateway=<i>gateway address</i> Temporary gateway address of the device during manufacturing.</p>
Example	<pre>connectprog /reboot /destip=192.168.1.1</pre>
See also	<ul style="list-style-type: none"> "Parameters used as flags" on page 38 for /reboot and other flag descriptions. "Digi Connect Programmer error messages" on page 48.

/? and /help - Display help text

Purpose Displays help text for the Digi Connect Programmer.

Syntax

```
connectprog /?
```

Or

```
connectprog /help
```

Parameters None

Example

```
connectprog /?
```

```
connectprog /help
```

Digi Connect Programmer error messages

Error messages are displayed when running Digi Connect Programmer interactively.

Error message	Reason and resolution
Failed to get device information.	Failed to connect to the device. This could be because the device is not currently present on the network.
Device does not have an IP address.	The device has a IP address of zero. IP address needs to be set before file upload requests can be done.
Failed to connect to device. Verify that the device settings, including the username/password are valid.	Failed to connect to the device. The most common cause is an authentication issue. Verify that the username and password are valid.
Failed to reboot device.	The reboot request was not successful.
Device not found.	With /mac=auto option, no device was detected on the network or /destip , no device was found with the specified IP address.
Failed to delete file system.	When /erasefs flag is used, an error occurred attempting to delete file system. The most common cause is an invalid username/password.
Failed to delete customization files.	When /erasecustom flag is used, an error occurred while attempting to delete customization files. The most common cause is an invalid customization username/password.
Configuration file not found.	The specified file does not exist. Verify path and file name.
Failed to upload file to device: The file wasn't found.	The specified file does not exist. Verify path and file name.
Failed to upload firmware to device. Check filename.	The specified file does not exist. Verify path and file name.
Failed to allocate memory.	Returned by /upload command. Unable to allocate enough PC memory for temporary storage of the file to be uploaded.
Failed to read configuration file.	Returned by /upload and /config commands. An error occurred reading the configuration file.
Failed to upload configuration to device.	An error while transferring the file to the Digi device.
No files found to upload.	Returned by /upload /dir if the specified directory was empty.
Failed to upload file to device.	Returned by /upload /file if there was an error while transferring the file to the device.
Failed to restore factory defaults.	Returned by /defaults command if error occurred while attempting to restore the device's factory defaults. Most common cause is an invalid username/password.

Digi Connect Programmer error codes

Error codes are displayed when running Digi Connect Programmer from a script.

Error code	Error message
0	Success
1	Error: File not found.
2	Error: Device not found.
3	Error: IP address not changed
4	Error: Couldn't connect to the destination device
5	Error: Couldn't write to the log file
6	Error: Memory couldn't be allocated for the request
7	Error: File not uploaded
8	Error: File not deleted
9	Error: Firmware update failed
10	Error: Can't reboot the device
11	Error: Invalid command request
12	Error: Query command failed
13	Error: Couldn't create the output file
14	Error: Couldn't write the output file
15	Error: Backup command failed
16	Error: Timeout waiting for a response
17	Error: Capability not supported by the destination device
18	Error: Invalid configuration file

This chapter shows how to perform customization and integration tasks.

Customizing Digi device look-and-feel

How do I customize and save changes to the web interface?

The recommended process for changing the look of the web interface is to customize the user interface using a single Digi device and the **Customization Administration** web page, verify that the customized web interface looks and operates as desired, then upload the customized interface to other devices using the Digi Connect Programmer.

Here are the steps you would follow to customize a the web interface for a factory default unit, and optionally, change the factory defaults.

1. Start with a factory default unit.
2. If a different stylesheet will be used, modify and upload the new stylesheet:
 - For multiple-user Digi devices log into the Digi device using the customization username and password. Under **Administration** on the left side of the page, click **Customization**.
For single-user Digi devices, enter **http://<ip address>/admin/customization.htm** and login as needed.
The **Customization Administration** page is displayed.
 - Download the stylesheet, a file named **config.ini**. On the **Customization Administration page**, click **Download Stylesheet Template (filename.css)**.
 - Open the stylesheet and make any changes as required.
 - Save and close the stylesheet file.
 - Login to the Digi device again and access the Customization Administration page.
 - In the **Upload File** edit box, enter the path and filename of the changed stylesheet file and click **Upload**.
3. If a different logo will be used, enter the name of the file containing the logo in the **Upload File** edit box, and click **Upload**.

4. As needed, customize company and product information displayed in the web interface.
 - Download the Configuration File Template, which is a file named **config.ini**. On the **Customization Administration** page, click **Download Configuration File Template (config.ini)**.
 - Open **config.ini**.
 - Change the **Company** section, **Product** section, and **ProductLinks** sections, see section descriptions on page 19.
 - If a new logo was uploaded (see step 2), specify the name of the file that was uploaded in the **Resources** section's **ProductLogo** setting
 - If a new stylesheet was uploaded (see step 3), specify the name of the file uploaded in the **Resources** section's **ProductStylesheet** setting.
 - Save and close **config.ini**.
 - Upload the config.ini file. On the Customization Administration page, in the **Upload File** edit box, enter path and filename for the config.ini file, and click **Upload**.
5. To use a different factory defaults than the Digi defaults, see page 53.
6. Reboot the device.
7. Open the web interface for the device. Verify the user interface has been customized as desired, and adjust settings as needed.
 - If you uploaded a customized web interface, but are still seeing the default one, clear the browser cache and then reopen the web interface.
 - Verify that the **ProductLogo** and **ProductStylesheet** entries in the config.ini file specify the correct filenames and that those logo and stylesheet files have been uploaded to the device.
8. The product should now be customized to your needs.

To load this customized web interface into other devices, see page 51.

If necessary, revert a customized device to the Digi-assigned default settings; see page 58.

How do I upload the customized web interface to other devices?

The most efficient way to upload the new web interface to other devices is to use the Digi Connect Programmer **/upload** command; see command description on page 41.

How do I change the customization username and password?

This procedure applies to Digi devices with a multiple-user model only. See "Determining the user model for your Digi device" on page 14 if you do not know the user model.

Multiple-user Digi models have a special username and password for logging on to the **Customization Administration** page. The default is **custom/custom**. This user definition is for customization activities only. It cannot be used to log in to the web interface or as the username and password when logging on to the command line interface. It should, however, be entered as a username and password for external tools, such as the Digi Connect Programmer, when uploading customization files.

Changing the username and password for logging on to this page is recommended to prevent unauthorized downloading and uploading of customization files, particularly the custom factory defaults.

Change the customization username and password

1. Log in to the Digi device, using the customization login.
2. On the Customization Administration page, click **Change Password**.
3. Change username, if desired.
4. Enter the new password in the **New Password** edit box, and again in the **Confirm Password** edit box.
5. Click **Apply**.
6. Changes are stored in a file called **passwd.ini**. To upload these changes and any other customization files needed using the Digi Connect Programmer **/upload** command; see command description on page 41.

Important: Record changed customization username and password

Remember the changed customization username and password and note them in your records. If you lose the changed username and password, password, there is no method to reset the Digi device back to the default . The unit must be sent back to Digi for remanufacturing; service fees apply.

How do I change default settings in a device (custom factory defaults)?

Custom factory defaults include the settings that take effect whenever a user resets their device to factory defaults. Custom factory defaults are useful when the Digi-assigned settings are not ideal for the device as used in your company's applications and solutions. Custom factory defaults are stored in a file named **defaults.rci**, which can be uploaded to other devices during manufacturing.

Use caution when changing default settings

Exercise caution in changing settings from Digi-assigned defaults, as changing certain settings can render the device inaccessible. For example,

- Disabling ADDP can cause the device to not be discovered on a network, even if it is actually connected.
- Disabling Telnet or HTTP services disables access to the device through the command-line or web interfaces.
- Changing network ports for network services such as ADDP, Telnet and HTTP from their default port numbers can cause Digi applications such as Device Discovery and Digi Connect Programmer to not complete their operations.

When changing device settings from Digi-assigned defaults, verify that the device operates appropriately with the changed defaults.

IP addresses and custom factory defaults

Using DHCP to assign IP addresses is recommended instead of assigning static IP addresses. If the device used for customizing factory defaults is configured with a static IP address, uploading the custom factory defaults file **defaults.rci** to other devices causes that same IP address to be assigned to all the devices to which the file is uploaded.

Create and use custom factory defaults

Here is the recommended procedure for creating and using custom factory defaults for your Digi devices:

1. Start with a device containing all the Digi-assigned defaults. Various device interfaces have an option to reset the device to factory defaults.
2. Modify defaults in the device, using one of the device interfaces, such as Digi Device Setup Wizard, web interface, or command-line interface.
3. Verify that the device operates appropriately with the changed defaults.
4. Open the web interface for the Digi device and display the **Customization Administration** page.

For multiple-user Digi devices log into the Digi device using the customization username and password. Under **Administration** on the left side of the page, click **Customization**.

For single-user Digi devices, enter

http://<ip address>/admin/customization.htm and login as needed.

The **Customization Administration** page is displayed.

5. Click **Download Custom Factory Defaults Template (defaults.rci)**. This file contains device settings established in step 2. This file is downloaded so that it can be uploaded to other devices. Direct editing of this file is not recommended without a thorough understanding of Remote Command Interface.
6. Upload file defaults.rci to a second device. This step tests using the custom factory defaults on another device.
 - Login to the other Digi device and access the **Customization Administration** page, as described in step 4.
 - In the **Upload File** edit box, enter file **defaults.rci** and the path to the file, and click **Upload**.
7. Reboot the second device. A reboot is required after uploading this file for its changes to take effect the next time a factory reset occurs.
8. Perform a factory reset. The device settings in the second device should be the custom factory defaults you established in step 2.
9. Upload **defaults.rci** to other Digi devices. The most efficient way to upload this file to multiple device is to use the Digi Connect Programmer's **/upload** command; see page 41.

Revert to Digi-assigned factory defaults

If necessary, revert a device from the custom factory defaults to the Digi-assigned factory defaults by deleting **defaults.rci** from the device's file system. See page 58.

How do the web interface and applet differ? Which should I use?

The web interface is an HTML-based interface provided by default for Digi products. The configuration applet is supplied as a sample alternate user interface for a Digi products, using Java and the Remote Command Interface (RCI) protocol. The applet does not have all of the functionality and options provided in the web interface. However, as a Java applet, it provides flexibility not possible in the web interface, such as implementing portions but not all of the device-interface functionality, in a customized form.

The web interface application resides on your PC and is executed from the PC. In contrast, the configuration applet resides on the Digi device, and is downloaded and run from within a browser on your PC.

The sample configuration applet and its accompanying documentation exposes the code base for the configuration applet shipped with Digi devices. This material demonstrates creating custom Java applications to configure and control the device.

The configuration applet is available for Digi Connect Family products only. It is not available for Digi Cellular Family products.

To view a document describing the sample configuration applet, choose **Start > Programs > Digi Connect > Samples > Java Applets > ConfigApp documentation**.

To view the source files for the configuration applet, choose **Start > Programs > Digi Connect > Samples > Java Applets > ConfigApp documentation**.

How do I create an applet?

You can create a new applet if your implementation requires one. What you are more likely to do is modify the source code for the sample configuration applet. There are several ways to customize the sample configuration applet, from simply editing a configuration file to changing or writing new Java code.

- To customize applet look-and-feel, edit file **config.ini**. Some settings are global and affect the entire user interface; others are specific to one view. For details on these settings, see the ConfigApp documentation (in the Digi Connect Integration Kit menu, **Start > Programs > Digi Connect > Samples > Java Applets > ConfigApp documentation**).
- To change text displayed in the applet user interface, such as labels for fields and message text, edit the resource **.properties** files.
- To customize views and content of the applet user interface, edit the **.java** source files.

To run and test the applet, load all necessary files onto the Digi device, as shown in the next topic.

How do I upload an applet?

To upload an applet, load the **.jar**, **.htm**, and **.ini** files onto the device. This only needs to be done once. When executed, the configuration applet connects to the device from which the applet was loaded.

1. Access the administration web interface by opening a browser and entering this URL:

`http://ip-address-of-device/admin/administration.htm`

2. In the **Administration** menu, select **File Management**. The File Management page is displayed.
3. Upload the necessary files for the applet, including the **.jar** files and **index.htm** files that reference the applet.

Load file **common.jar**:

- On the File Management page, click **Browse**.
- Locate and select the file named **common.jar** in the **configapp\dist\bin\applet** directory.
- Click **Open**.
- Click **Upload**.

Repeat these steps for file **configapp.jar**, **index.htm**, and **config.ini**.

4. To run the configuration applet from the device, click **index.htm**.

To access the applet directly, use this URL:

`http://ip address of device/FS/WEB/index.htm`.

5. To use the applet as the default interface for the device, return to the Home page of the default web interface. In the **User Interfaces** section, on the line for **Custom Interface**, click the **Set as Default** button. This action creates a hidden file named **.default**.

How do I revert a customized device to the Digi-assigned defaults?

Occasionally it is desired to revert a customized Digi device to the Digi-assigned default settings. To do this, remove the customization files uploaded to the device and reboot the device. The files removed depend on whether the web or configuration applet interface is used.

If device is using web interface and not configuration applet

1. Use the Digi Connect Programmer **/upload** command, with the **/erasecustom** parameter. This command erases all customization-related files in the Digi device.
2. Reboot the device.

If device has been loaded with configuration applet

Use the Digi Connect Programmer **/upload** command, with the **/erasefs** parameter. This command erases the applet files from the Digi device.

Manufacturability

How can I efficiently configure and deploy devices?

Using the Digi Connect Programmer, you can load configuration and customization files into multiple Digi devices. This is more efficient than uploading files to devices one at a time through the web interface. In addition, you can create scripts of Digi Connect Programmer commands and other commands to handle the firmware loading, device configuration, and other operations in a sequenced manner. The sample manufacturing environment in the Integration Kit provides examples of such scripts.

How do I upload a customized web interface to other devices?

The most efficient way to upload the new web interface to other devices is to use the **/upload** command and the **/config=filename** and **/erasecustom** parameters. See page 41.

The web interface configuration and customization files to upload include the **config.ini** file, custom logo and stylesheet files, and, for multiple-user models of Digi devices, the **passwd.ini** file, a file created if you changed the customization username and password from their defaults.

How do I upload custom factory defaults to multiple devices?

Use the **/upload** command and the **/custom=filename.rci** parameter.

How do I upload an applet to multiple devices?

Use the **/upload** command and the **/dir=directory**, **/file=filename**, and **/erasefs** parameters.

How do I load a preferred firmware image during manufacturing?

Use the **/upload** command and the **/firmware=filename** parameter.

What do I need to set up a manufacturing environment?

A manufacturing environment is the hardware and software equipment and tools involved in configuring Digi devices with a customized look-and-feel and operation. Typically, this manufacturing environment would include:

- A manufacturing PC, executing applications and scripts for configuring devices.
- Digi devices to be configured with custom user interface software and custom factory defaults.
- A private network, with Dynamic Host Configuration Protocol (DHCP) enabled for IP address assignment. For wireless devices, a DHCP server for assigning IP addresses is particularly useful. A DHCP server is available at the Web site for Tftp32 at this location:

<http://perso.wanadoo.fr/philippe.jounin/tftpd32.html>

- Digi Connect Programmer
- A set of scripts running on the manufacturing PC. The scripts perform a variety of operations, such as displaying prompts for user input on the manufacturing PC, using ADDP to discover devices, and executing Digi Connect Programmer commands to load customization files.

A sample manufacturing environment is included in the Digi Connect Integration Kit. It includes scripts and code samples for configuring Digi devices and uploading customization files. See the readme file and comments in the scripts for more information on the operations performed.

Implementing software interfaces

What software interfaces are available on Digi devices?

Digi devices support several common interfaces to devices, such as Telnet, raw TCP sockets, UDP sockets, RealPort, HTML, FTP, SSL, and SNMP. These common interfaces are discussed in the *User's Guide* for your Digi device.

In addition to these common interfaces, there are several Digi-proprietary interfaces:

- Advanced Digi Discovery Protocol (ADDP), a Digi International-proprietary protocol for discovering devices on networks.
- Remote Command Interface (RCI), a programmatic interface for configuring and controlling Connect family devices.

Device configuration and monitoring through Simple Network Management Protocol (SNMP) involves both standard and Digi-specific Management Information Blocks (MIBs).

Also, there are elements of the standard Digi devices that can be redeployed in your company's software distribution, including RealPort drivers and Digi device documentation.

How does device discovery work, and how can it be modified?

Device discovery is a method of identifying Digi devices connected to a network.

What is Advanced Digi Discovery Protocol (ADDP)?

Advanced Digi Discovery Protocol (ADDP) is a Digi International-proprietary protocol for discovering devices on networks. ADDP runs on any operating system capable of sending multicast IP packets to a network. ADDP sends out a User Datagram Protocol (UDP) multicast packet to all devices on the network. Devices that support ADDP reply to this UDP multicast with their configuration information. Even devices that do not yet have an IP address assigned, or that are misconfigured for the subnet can reply to the UDP multicast packet.

ADDP needs to communicate with the Transmission Control Protocol/Internet Protocol (TCP/IP) stack using UDP. The TCP/IP stack should be able to receive multicast packets and transmit datagrams on the network.

Which ports does ADDP use?

ADDP uses port **2362**, which is a UDP port. This means that the port will not respond to any queries other than UDP ones.

What are my options for interfacing with ADDP?

ADDP functions are supplied in two formats, providing flexibility in how they are implemented in applications:

- As static library (**.lib**) files. The ADDP functions are directly supplied in your application files, and you do not have to ship extra files. The drawback to using these files is that there is limited flexibility. The functions can only be linked to Microsoft Visual C++ applications.
- As Dynamically Linked Library (**.dll**) files associated with the application. Using these files allows for more flexibility, as the functions can be linked to interfaces other than Microsoft Visual C++ applications. However, this approach also means that more files must be associated with the main application at runtime.

How can I modify the sample discovery program (ADDP)?

Digi owns and maintains ADDP. While you cannot change the protocol itself, you can change how ADDP is called, and how ADDP's discovery functions display results. The Microsoft Visual C++ version of ADDP is supplied with the Digi Connect Integration Kit, along with an HTML help file describing it. You can modify this sample or create your own discovery program, using most common Windows programming languages.

How can I discover and configure devices from VisualBasic® or other languages?

Digi has sample code and VisualBasic module that you can include in your project for. This sample code is written for VisualBasic 6.0. To receive the VisualBasic code, or for help with other versions of VisualBasic or other languages, send an email request to this address:

connectme@digi.com

What other Digi-supplied libraries or DLLs are available for use in my applications?

The Remote Command Interface (RCI) library is available for use in applications.

What is the Remote Command Interface (RCI)?

Remote Command Interface (RCI) is a programmatic interface for configuring and controlling Digi devices. RCI is Digi-proprietary XML-based request/response protocol that allows a caller to query and modify device configurations, access statistics, reboot the device, and reset the device to factory defaults.

Unlike other configuration interfaces that are designed for a user, such as the web and command-line interfaces, RCI is designed to be used by a program. Example uses of RCI include custom interface in Java applet format and stored on the Digi device, and custom application running on a PC that monitors and controls installation of many Digi devices.

For more details on using RCI in your applications, see "Remote Command Interface (RCI)" on page 29 and the *Remote Command Interface (RCI) Specification*.

Which port does RCI use?

RCI uses a default port of **80**. This is the standard port used with HTTP. Using this port allows RCI to tunnel through firewalls without requiring a special port to be defined.

Can I configure and monitor devices using SNMP?

Yes. Simple Network Management Protocol (SNMP) is a protocol for managing and monitoring network devices. SNMP architecture enables a network administrator to manage nodes--servers, workstations, routers, switches, hubs, etc.--on an IP network; manage network performance, find and solve network problems, and plan for network growth. Digi devices support SNMP Version 1.

Can I redeploy RealPort drivers?

Yes. You can include Digi RealPort drivers in your own company's software distribution. The most recent versions of RealPort drivers are on the Digi Support site (<http://www.digi.com/support/>) and on the Microsoft update site (<http://update.microsoft.com>).

To rename or customize the RealPort driver files, for example, to private-label them, you must contact Digi International for assistance.

How can I customize the product documentation?

The standard product documentation for Digi devices includes:

- *A User's Guide* for the Digi device's product family
- *Digi Connect Family Command Reference* (applies to all product families listed on the cover of this guide)
- *Remote Command Interface (RCI) Specification*
- *Hardware References* for Digi Connect Family products, provided for some but not all device families

If you need to make significant changes to the user documentation to reflect device customization efforts, Digi International can provide you with the source for these documents. For example, you may wish to create a custom Command Reference or User's Guide that covers only a single product, or you may want to merge content from these documents into your own company's product documentation.

Most source documents are produced using Adobe® FrameMaker®.

Implementing hardware interfaces

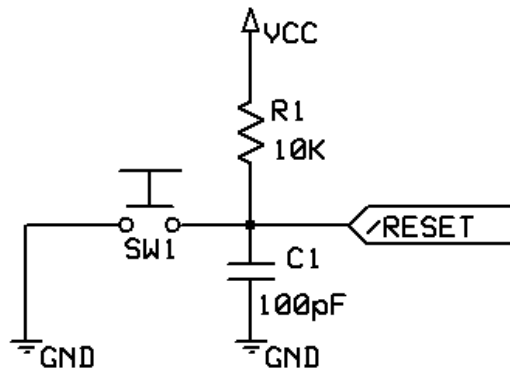
How should I implement reset functions on my Digi device?

There are two kinds of reset functions for a Digi device:

- Hard reset
- Soft reset

Hard reset

Digi Connect embedded modules support a hard reset, which resets the entire device. The pin number for the hard reset varies by device. The device is forced into a hard reset when this pin is pulled low.



Soft reset

A soft reset is a reset controlled by the firmware rather than hardware. There are two kinds of soft reset indications used in Digi devices running standard plug-and-play firmware. Most Digi devices have a push button that is tied to the soft reset signal. The Digi Connect ME has a pin that needs to be grounded. When the pin is grounded, it is exactly equivalent to pushing and holding the button. The rest of this discussion assumes a push button being used.

There are two effects to pushing the reset button:

- A simple "reset." The device resets as if power cycled, using the currently saved device configuration settings.
- A "reset to factory defaults." This type of reset occurs if you power off the unit, then hold in the button while you power it on and keep the button depressed until the unit flashes 1-5-1 (the only flash that is really significant). That indicates that the unit has reset its nonvolatile storage to defaults. It will keep flashing 1-5-1 until you release the button, at which point the unit will continue to boot.

If, after the unit is running, you press the button for a second and then release it, the device will reboot.

How can I use RCI over serial?

Remote Command Interface (RCI) is a proprietary protocol developed by Digi International. RCI can be used over Ethernet or serial.

The intention behind RCI over serial is to provide a mechanism for embedded hosts to control the configuration of a Digi device via the only data interconnect between the two; namely, the serial port. One of the serial pins is allowed to become a control signal, telling us that data being received on the serial port should be interpreted as RCI queries. The standard XML formatted RCI responses are returned to the embedded host via the serial port for parsing, if necessary.

Using RCI over serial requires these steps:

1. The configuration of the Digi device must be changed, through some non-RCI-over-serial means, so that RCI over serial is enabled as an available feature for a specific serial port. What this really means is that the DSR signal is no longer a serial signal, but is instead an indicator from the embedded host that the RCI over serial mode is desired.
2. Assert the DSR signal. The unit, assuming the serial port is not already in use, will detect the signal change and switch into the RCI over serial mode, and send an XML-formatted announcement. To more predictably enter the RCI over serial state, one would power our unit down, assert the DSR signal, then power the unit on with the DSR signal asserted. This will guarantee that the RCI over serial behavior will be the first functionality granted to the serial port.
3. While in RCI over serial mode, the serial port will be 9600 baud, 8 data bits, 1 stop bits, no parity.
4. Submit a properly XML formatted RCI request (that is, XML beginning with <rci_request>...) via the serial port.
5. The unit will parse the request, then send a response back out of the serial port.
6. If the unit is not in the middle of processing a request, de-asserting the DSR signal will return the serial port to normal operation. For a more predictable exit from the RCI over serial mode, one would power down the unit, de-assert the DSR signal, and power on the unit.

For more information, see the *Remote Command Interface (RCI) Specification*, in the RCI over Serial section.

/? command 47

A

ADDP

See Advanced Digi Discovery Protocol

Advanced Digi Discovery Protocol (ADDP)

- discovering/configuring devices from VisualBasic or other languages 62
- modifying sample program 62
- options for interfacing with 62
- overview 28, 61
- port number for 61
- use in /discover command 39

applet

- comparison with web interface 55
- creating 56
- files for 23
- overview 22
- setting as default user interface 57
- uploading files to devices 16, 24, 57

authentication file for customization username and password 15

C

COM port redirection 32

common.jar file 23, 57

Company section in config.ini 19

config.ini file

- for configuration applet 23, 24
- for web interface 13, 16

configapp.jar file 23, 57

configuration applet

- .default file 23, 24, 57
- comparison with Web user interface 55
- customizing 24
- files 23
- overview 22
- setting as default user interface 24, 57
- uploading files 24

Configuration File Template (config.ini) 13, 16

configuration interfaces

- Java applet interface 22

connection failure for device 48, 49

connectprog command 36

Copyright setting 19

.css files (custom stylesheet) 21

custom factory defaults

- caution on disabling services 21
- creating and using 53
- defaults.rci file 21
- failure to restore 48
- overview 21
- template file for 13, 21
- uploading to devices 41

custom/custom username and password 15, 52

customization

- company information in user interface 7, 19
- downloading files for 15
- links to product information in user interface 20
- logos 20

- of factory defaults 7, 21, 53

- overview 7

- product documentation 64

- product information in user interface 7, 20

- product look-and-feel 7, 50

- stylesheets 20

- username and password for 52

Customization Administration page

- accessing 14

- changing username and password for 15

- downloading files 15

- overview 13, 14

- uploading files 15

customization files

- erasing from device 58

- uploading to devices 41

customization username and password

- authentication file (passwd.ini) 13

D

.default file 23, 24, 57

default gateway 40

default port numbers

- for Advanced Digi Discovery Protocol (ADDP) 61

- for Remote Command Interface (RCI) 63

/defaults command 45

defaults.rci file 13, 15, 21, 38, 42, 53, 54

device discovery

- description 61

- See also Advanced Digi Discovery protocol

device not found error 48, 49

DHCP

- See Dynamic Host Configuration Protocol

Digi Connect Programmer

- /defaults command 45

- /discover command 39

- /erasecustom parameter 38

- /erasefs parameter 38

- /pause parameter 38

- /reboot command 46

- /reboot parameter 38

- /set command 40

- /silent parameter 38

- /upload command 41

- basic usage rules 36

- connectprog command 36

- discovering devices on a network 39

- error messages 48

- help text for 47

- limitations of 36

- parameters 37

- parameters used as flags 36, 38

- rebooting devices 46

- reverting device to factory defaults 45

- setting network parameters 40

- syntax rules 36

- uploading files to devices 41

- username and password specification on commands 36

- using commands in scripts 48
- Digi contact information 6
- DIGI-DEVICE-INFO.mib 31
- /discover command 39
- displaying help text for Digi Connect Programmer 47
- documentation
 - customizing 64
 - URL setting in config.ini file 20
- DocumentationURL setting 20
- downloading files
 - configuration file template (config.ini file) 15
 - custom factory defaults template (defaults.rci file) 15
 - custom stylesheet (.css) template 15
- Dynamic Host Configuration Protocol (DHCP) 27, 60

E

- /erasecustom parameter 38, 58
- /erasefs parameter 38
- erasing customization files from a device 58
- Extensible Markup Language (XML) 29

F

- factory defaults
 - custom 21, 41
 - failure to restore 48
 - reverting to 45
- failures
 - configuration file not found 48, 49
 - deleting customization files 48
 - deleting file system 48
 - device connection attempt 48, 49
 - device reboot 48
 - getting device information 48
 - memory allocation 48
 - no files found to upload 48
 - reading configuration file 48
 - restoring factory defaults 48
 - uploading configuration to device 48
 - uploading files 48
 - uploading files to device 48
 - uploading firmware 48
- file and file-system uploading
 - /upload command 41
 - error messages 48
- File Management page
 - uploading applet files 16
- firmware images
 - preferred image for uploading 24
 - uploading 41
 - URL for firmware updates in config.ini file 20
- FirmwareURL setting 20

H

- hardware interfaces
 - RCI over serial 33, 66
 - reset functions 33
- /help command 47

I

- index.htm file for configuration applet 23, 57
- INI files 18
- integration
 - examples 8
 - hardware 8

- overview 7
- software 8
- IP address
 - none (0) for device 48
 - setting through the /set command 40
 - setting using DHCP 27, 53, 60
 - specifying on Digi Connect Programmer commands 36

J

- Java applet interface
 - as a device configuration interface 22
 - developing custom applets 22
- .java source files 23, 24

L

- logo
 - customizing 20
 - Digi company logo (logo.gif) 20
- LongName setting 19

M

- MAC address 36
- Management Information Base (MIB)
 - DIGI-DEVICE-INFO.mib 31
 - MIB-I 31
 - MIB-II 31
- manufacturability
 - efficient device configuration and deployment 59
 - manufacturing environment 60
 - scenarios 59
 - tools and resources 24
 - uploading an applet to multiple devices 59
 - uploading custom factory defaults to multiple devices 59
 - uploading customized Web UI to devices 59
 - uploading preferred firmware image 59
- memory allocation failure 48
- MIB-I 31
- MIB-II 31

P

- passwd.ini file 13, 15
- /pause parameter 38
- preferred firmware image 24
- Product section in config.ini 20
- ProductClass setting 20
- ProductLinks section in config.ini 20
- ProductLogo setting 20
- ProductName setting 20
- ProductStylesheet setting 20
- Python files
 - uploading 42

R

- RCI
 - See Remote Command Interface
- RCI over serial 66
- RealPort
 - overview 32
- RealPort drivers
 - redeploying 63
- /reboot command 46
- reboot failure 48
- /reboot parameter 38
- rebooting devices 46
- redeployable elements

- product documentation 64
- RealPort drivers 63
- Remote Command Interface (RCI)
 - as a device configuration interface 29, 63
 - as a device interface 10
 - custom factory defaults file 21
 - overview 63
 - port number for 63
- reset functions
 - hard reset 65
 - reset to factory defaults 65
 - simple reset 65
 - soft reset 65
- resource .properties files 23, 24
- Resources section in config.ini 20
- reverting devices
 - to custom factory defaults 21, 53
 - to Digi-assigned factory defaults 45, 54, 58
- RFC 1213 31
- RFC 1316 31
- RFC 1317 31

S

- /set command 40
- setting network parameters 40
- ShortName setting 19
- /silent parameter 38
- Simple Network Management Protocol (SNMP)
 - as a device configuration and monitoring interface 63
 - as a device configuration interface 30, 63
 - as a device-monitoring interface 31
 - monitoring capabilities 31
- SNMP
 - See Simple Network Management Protocol
- software interfaces
 - Advanced Digi Discovery Protocol (ADDP) 28, 61
 - overview 28, 61
 - RealPort drivers 32, 63
 - Remote Command Interface (RCI) 29, 63
 - scenarios 61
 - Simple Network Management Protocol (SNMP) 30, 63
- statistics
 - available through SNMP 31
 - device information 31
 - network 31
 - port 31
- Strings section in config.ini 20
- stylesheets
 - config.ini setting 20
 - Digi default (stylesheet.css) 20
 - template file 21
 - template for (.css) 13
- subnet mask 40
- support, Digi contact information 6

T

- TechSupportURL setting 20
- TTY port redirection 32

U

- /upload command 41
- uploading files
 - applet files 16
 - config.ini file 15
 - custom factory defaults (defaults.rci file) 41

- custom factory defaults (defaults.rci) 15
- custom stylesheets (.css) 15
- defaults.rci file 15
- failures and error messages for 48
- file-system files 41
- firmware images 42
- from the Customization Administration page 15
- from the Digi Connect Programmer /upload
 - command 41
- Python files 42
- Web customization files 41

V

- vendor ID 39
- /vendorid option 39
- VisualBasic, device discovery and configuration
 - from 62

W

- web interface
 - comparison with configuration applet 55
 - custom stylesheet for 15
 - customization tools for 13
- WebSiteName setting 19
- WebSiteURL setting 19

X

- XML
 - See Extensible Markup Language (XML)