



# RFID Anywhere™ Getting Started Guide

Part number: DC00255-01-0200-01  
Last modified: February 2006

---

Copyright © 2006 iAnywhere Solutions, Inc. Portions copyright © 2006 Sybase, Inc. All rights reserved.

iAnywhere Solutions, Inc. is a subsidiary of Sybase, Inc.

iAnywhere grants you permission to use this document for your own informational, educational, and other non-commercial purposes; provided that (1) you include this and all other copyright and proprietary notices in the document in all copies; (2) you do not attempt to “pass-off” the document as your own; and (3) you do not modify the document. You may not publish or distribute the document or any portion thereof without the express prior written consent of iAnywhere.

This document is not a commitment on the part of iAnywhere to do or refrain from any activity, and iAnywhere may change the content of this document at its sole discretion without notice. Except as otherwise provided in a written agreement between you and iAnywhere, this document is provided “as is”, and iAnywhere assumes no liability for its use or any inaccuracies it may contain.

iAnywhere<sup>®</sup>, Sybase<sup>®</sup>, and the marks listed at <http://www.iAnywhere.com/trademarks> are trademarks of Sybase, Inc. or its subsidiaries. <sup>®</sup> indicates registration in the United States of America.

Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

# Contents

<b>About This Manual</b>	<b>vii</b>
RFID Anywhere documentation . . . . .	viii
Finding out more and providing feedback . . . . .	ix
<b>I Using RFID Anywhere</b>	<b>1</b>
<b>1 Getting Started with RFID Anywhere</b>	<b>3</b>
Introduction to RFID Anywhere . . . . .	4
RFID Anywhere security overview . . . . .	13
Installation instructions . . . . .	15
Post-install analysis . . . . .	26
Uninstalling RFID Anywhere . . . . .	29
<b>2 The Administrator Console</b>	<b>31</b>
Using the Administrator Console . . . . .	32
The Login page . . . . .	33
The Home page . . . . .	34
RFID Anywhere licensing . . . . .	42
Monitoring an RFID Anywhere network . . . . .	43
The Provisioning page . . . . .	47
The Configuration page . . . . .	51
Editing local security . . . . .	53
Editing domain security . . . . .	55
<b>3 Default RFID Anywhere Services</b>	<b>57</b>
Default local services . . . . .	58
<b>4 The RFID Simulator Data Editor</b>	<b>79</b>
Using the RFID Simulator Data Editor . . . . .	80
Working with simulator files . . . . .	82
Summary . . . . .	87
<b>5 The Label Designer</b>	<b>89</b>
Using the Label Designer . . . . .	90
How label designs are used in RFID Anywhere . . . . .	93
<b>6 Getting Started with Data Protocol Processors (DPPs)</b>	<b>95</b>
DPP overview . . . . .	96

---

DPP flexibility . . . . .	97
Using DPPs . . . . .	98
Summary . . . . .	99
<b>7 RFID Anywhere Messaging Connectors</b>	<b>101</b>
Introduction . . . . .	102
Adding and configuring connectors . . . . .	104
Installing future connectors . . . . .	105
File messaging connector . . . . .	106
MSMQ messaging connector . . . . .	110
QAnywhere messaging connector . . . . .	113
SMTP messaging connector . . . . .	118
TCP messaging connector . . . . .	121
UDP messaging connector . . . . .	124
Web Service messaging connector . . . . .	127
Troubleshooting . . . . .	130
<b>II RFID Anywhere Tutorials</b>	<b>133</b>
<b>8 About the Tutorials</b>	<b>135</b>
<b>9 Creating Simulated RFID Data</b>	<b>137</b>
Introduction . . . . .	138
Lesson 1: Designing the Tag Activity timeline . . . . .	139
Lesson 2: Using the RFID Simulator Data Editor to create simulation data . . . . .	142
Lesson 3: Configuring the RFID Multiprotocol Simulator con- nector . . . . .	150
Summary . . . . .	152
<b>10 Using ALE to Filter and View RFID Data</b>	<b>153</b>
Introduction . . . . .	154
Lesson 1: Working with connectors from the Administrator Console . . . . .	160
Lesson 2: Starting the connectors . . . . .	168
Lesson 3: Viewing and understanding ALE reports . . . . .	169
Lesson 4: Stopping the connectors . . . . .	175
Properties of ALE Event Cycle connectors . . . . .	176
Summary . . . . .	181
<b>11 Using Report Engine MP</b>	<b>183</b>
Introduction . . . . .	184

---

Lesson 1: Working with connectors from the Administrator Console . . . . .	186
Lesson 2: Starting the demo . . . . .	189
Lesson 3: Viewing the simulation reports . . . . .	190
A sample multiprotocol report . . . . .	191
Lesson 4: Stopping the connectors . . . . .	193
Properties of RFID MultiProtocol Report Definition connectors	194
Summary . . . . .	201
<b>12 Inventory Tracker Demo</b>	<b>203</b>
Introduction to the Inventory Tracker demo . . . . .	205
Lesson 1: Setting up the simulator connectors . . . . .	209
Lesson 2: The ALE demo . . . . .	211
Lesson 3: The Report Engine MP demo . . . . .	216
Lesson 4: The business module demo . . . . .	221
Lesson 5: Using web services to access business module methods . . . . .	226
Lesson 6: Customizing the Inventory Tracker demo . . . . .	228
Installing the pre-configured Inventory Tracker demo . . . . .	232
Summary . . . . .	237
<b>Index</b>	<b>239</b>



---

# About This Manual

Subject	This manual describes RFID Anywhere, a platform for building RFID solutions. It provides instructions on installing and configuring RFID Anywhere, as well as tutorials that demonstrate how you can use RFID Anywhere to administer and test your RFID network.
Audience	This manual is for all users of RFID Anywhere.
Before you begin	This manual assumes an elementary familiarity with RFID networks.

---

# RFID Anywhere documentation

The RFID Anywhere documentation set

This book is part of the RFID Anywhere documentation set. This section describes the materials in the documentation set and how you can use them.

The RFID Anywhere documentation set consists of the following components:

- ◆ **RFID Anywhere Getting Started Guide** This manual describes RFID Anywhere, a platform for building RFID solutions. It provides instructions on installing and configuring RFID Anywhere, as well as tutorials that demonstrate how you can use RFID Anywhere to administer and test your RFID network.
- ◆ **RFID Anywhere Developer's Guide** This manual describes the RFID Anywhere Visual Studio Extension, and explains how to write custom components, such as business modules for your RFID Anywhere network. It also describes other RFID Anywhere development tasks.
- ◆ **RFID Anywhere Help** This collection of HTML files provides help for local services running in RFID Anywhere, and can be viewed from the Administrator Console Properties Manager. These files should be downloaded and extracted to your RFID Anywhere installation directory to make them available when configuring a connector or business module from the Administrator Console.

Visit [http://www.ianywhere.com/developer/product\\_manuals/rfid\\_anywhere/index.html](http://www.ianywhere.com/developer/product_manuals/rfid_anywhere/index.html) to obtain the latest version of the RFID Anywhere documentation.

---

## Finding out more and providing feedback

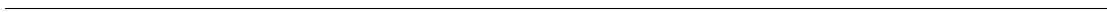
### Finding out more

Additional information and resources, including the RFID Anywhere Developer Community, are available at <http://www.ianywhere.com/RFID>.

### Feedback

We would like to receive your opinions, suggestions, and feedback on this documentation.

You can email comments and suggestions to the RFID Anywhere documentation team at [iasdoc@ianywhere.com](mailto:iasdoc@ianywhere.com). Although we do not reply to emails sent to this address, we read all suggestions with interest.



PART I

# USING RFID ANYWHERE

This part provides an introduction to RFID Anywhere and its components, and also includes system requirements and installation instructions.



---

## CHAPTER 1

# Getting Started with RFID Anywhere

About this chapter

This chapter provides an overview of RFID Anywhere and explains how to install it.

Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Introduction to RFID Anywhere</a>	4
<a href="#">RFID Anywhere security overview</a>	13
<a href="#">Installation instructions</a>	15
<a href="#">Post-install analysis</a>	26
<a href="#">Uninstalling RFID Anywhere</a>	29

## Introduction to RFID Anywhere

This section provides a high-level introduction to RFID Anywhere and explains the components of RFID Anywhere.

☞ For more information on the features and benefits of RFID Anywhere, see the online resources at <http://www.ianywhere.com/RFID>.

### RFID Anywhere overview

RFID Anywhere is a software platform that simplifies every phase of radio frequency identification (RFID) projects, including development, deployment, and ultimately the management of highly distributed, multi-site networks. RFID Anywhere manages the interfaces with RFID readers and printers, as well as bar code equipment, sensors, and controllers. As a result, developers can focus on writing business logic, not low-level device or network interfaces.

RFID Anywhere is built on the .NET platform, and provides a service-oriented architecture where multiple applications can consume processed data. This service-oriented architecture allows organizations to get the most out of RFID investments by allowing them to leverage the same RFID hardware to fuel any number of departmental or enterprise applications.

RFID Anywhere supports a broad array of readers, tag formats, and protocols, and can be deployed to the edge of the network, reducing the flow of data across the network and enabling real-time responses to RFID event data. Providing enterprise-class management and security for the entire solution, RFID Anywhere minimizes downtime and reduces the ongoing cost of RFID solutions.

RFID Anywhere insulates developers and systems integrators from the wide variety of low-level hardware interfaces, industry standards, and protocols in the RFID space. Hardware abstraction allows developers to focus on creating business logic and adding value to the vast amount of data produced by RFID networks. Processed data can then easily be integrated into enterprise applications through a variety of transports.

The plug-and-play architecture of RFID Anywhere has been designed to provide support for current hardware, standards, and protocols. When new hardware is introduced, or when new standards emerge, the architecture of RFID Anywhere enables them to be supported without requiring a new product release or complete reinstall.

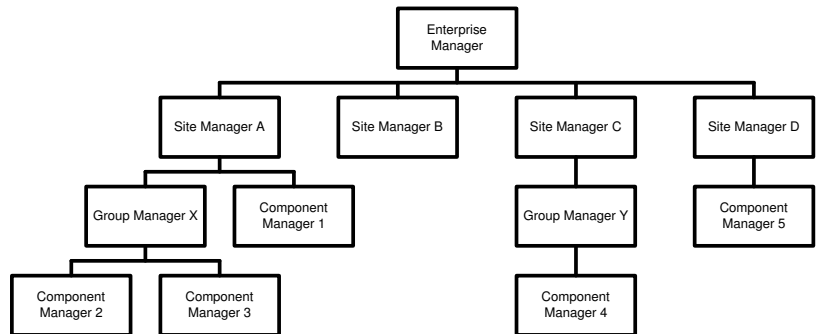
RFID Anywhere is embeddable technology that can implement the

functionality of a smart reader. With the ability to process data at the edge of an RFID network, only valuable information processed from raw RFID data needs to be sent to enterprise systems.

Developers and systems integrators will benefit from included tools, centralized network management, and easy to use interfaces, allowing for easy development and maintenance of powerful RFID solutions.

## A powerful distributed architecture

RFID Anywhere can be installed in one of three modes: as a **Component Manager**, as a **Group Manager**, or as a **Site Manager**.



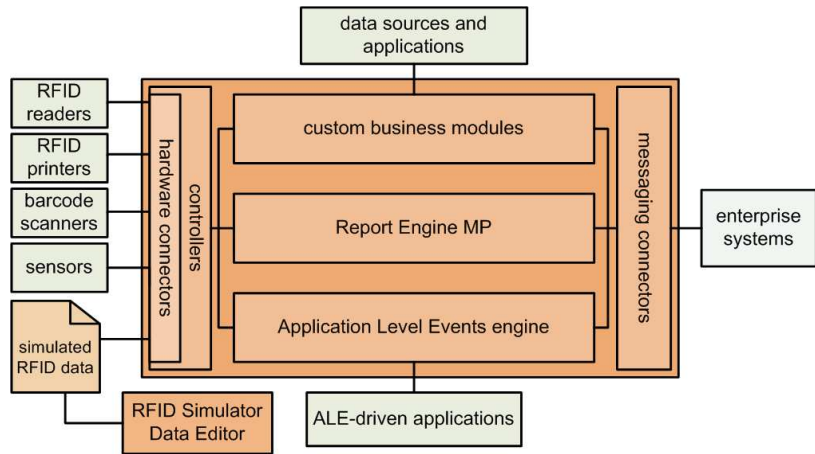
Component Managers are solely responsible for collecting data and sending it further up the RFID network for aggregation and processing. These nodes can perform filtering and grouping of data, but they do not run business logic. For these reasons, business modules are not and can not be installed on Component Managers—only connectors and controllers are installed.

Group Managers are able to run business logic. They can process data from connections running locally, and can process and aggregate data from Component Manager nodes underneath it. For example, Group Manager X in the above diagram can have its own connectors communicating with an RFID reader, and can consolidate that data with data from connectors running on Component Managers 2 and 3.

Site Managers are similar to Group Managers in that they can run business logic, they can host their own connectors and controllers, and can aggregate data from nodes beneath them. A Site Manager can also manage systems below it from the Site Manager's Administrator Console. They have the additional responsibility of communicating with an Enterprise Manager on behalf of the site. Enterprise Manager is an add-on component to RFID Anywhere that provides additional functionality for managing a distributed RFID network.

## RFID Anywhere components

This section provides an overview of the various components of RFID Anywhere.



### Hardware connectors

Hardware **connectors** installed in RFID Anywhere provide the interface to physical hardware devices. Each connector is built using the native, low-level hardware interfaces to communicate with a specific piece of hardware. Connectors expose configuration properties, send hardware commands, and receive data from these hardware devices. By abstracting the low-level details of a piece of hardware, RFID Anywhere provides a standard interface for interacting with hardware and for gathering raw data to pass to other RFID Anywhere components. This connector architecture thus shields developers and integrators from having to learn the low-level specifics of each piece of hardware, and allows them to easily support multiple types of hardware from their solution.

Each specific piece of hardware that RFID Anywhere supports has its own unique connector that can be installed in RFID Anywhere.

### Controllers

In RFID Anywhere, a **controller** exposes the functionality necessary to support a specific family of devices such as RFID readers, bar code scanners, RFID printers, or sensors. The functionality exposed by a given controller is common to all devices in the family of hardware that the controller supports. For example, the RFID reader connector captures and delivers RFID tag

events and exposes triggers to allow readers to be controlled by business logic. Business logic running in RFID Anywhere interacts with controllers, as controllers encapsulate the various hardware connectors installed on a system. This allows data from multiple different pieces of hardware to be seamlessly aggregated and used by a single application.

☞ For more information about controllers, see [“Understanding RFID Anywhere Controllers”](#) [*RFID Anywhere Developer’s Guide*, page 31].

## RFID Network Simulator

RFID Anywhere’s **RFID Network Simulator** takes the place of an expensive physical lab for the purposes of developing and testing business logic. By simulating the appearance and departure of a series of tags with a definable set of values, formats, and protocols, developers can check special cases and perform load testing without acquiring and installing physical RFID hardware. In addition, system integrators can save time and money with simulated proof-of-concept systems that can be developed and demonstrated anywhere.

All of this is accomplished with the RFID Simulator Data Editor and RFID Multiprotocol Simulator connector—an RFID Anywhere component that uses simulated data to feed raw RFID data into business logic. By using a connector, simulated RFID data is fed into RFID Anywhere in the exact same way as raw data coming from a reader.

Using this simulation environment, hardware investigation and decision making can be done in parallel with the software and business logic development tasks.

☞ For more information about simulating RFID networks with the RFID Network Simulator, see [“The RFID Simulator Data Editor”](#) on page 79.

## Custom business modules

Custom **business modules** are edge-processing components written with the .NET platform. Business modules get built and deployed to RFID Anywhere and perform the following primary tasks:

- ◆ Connecting to controllers to receive events such as RFID tag reads
- ◆ Issuing read triggers to hardware to begin collecting information
- ◆ Responding to hardware events and errors
- ◆ Querying and updating secondary sources such as databases and other applications

- ◆ Sending processed information to enterprise systems through messaging connectors

Developers are given control in each of the above areas to leverage the power of custom code at the edgware level. This is done through a common API, making the process straightforward.

The RFID Anywhere Visual Studio .NET Extension speeds the development process by automatically creating shell code for custom business logic. In addition, the extension builds and deploys these business modules that are automatically detected by RFID Anywhere. Custom business modules never require developers to write code or perform configuration relating to Windows services and other low-level tasks, and all development work is accomplished inside the Visual Studio environment.

This strong foundation allows developers to focus on writing business logic immediately rather than designing its supporting structure. Each business module runs in its own process to take advantage of domain isolation and to simplify development and testing.

Applications running outside of RFID Anywhere can also interact with custom business modules using Web Services.

☞ For more information about creating custom business modules, see “Using the RFID Anywhere Visual Studio Extension to Create Custom Business Modules” [*RFID Anywhere Developer’s Guide*, page 7].

## Report Engine MP

RFID Anywhere’s **Report Engine MP** provides a development model that allows developers and integrators to process raw RFID data by enabling the simple creation of XML reports based on the activity of tags from multiple protocols. These XML reports can then be fed into other enterprise systems by using RFID Anywhere’s messaging connectors.

While the ALE standard enables the generation of XML reports for tag activity, it is limited in that it can only generate these reports from EPC tags. This limitation is addressed with RFID Anywhere’s Report Engine MP by enabling the creation of XML reports based on tag activity for both EPC and ISO tags. The Report Engine MP decodes the tags of various protocols for the creation of consolidated reports.

☞ For more information about the Report Engine MP development model, see “Using Report Engine MP” on page 183.

## Application Level Events (ALE) engine

RFID Anywhere includes an **Application Level Events (ALE)** engine to

provide another means for developers and integrators to process raw RFID data. ALE is a standard defined by EPCGlobal, Inc. (<http://www.epcglobalinc.org>) that specifies the simple creation of standard XML reports based on the tag activity of EPC tags. These XML reports can then be fed into other enterprise systems by using RFID Anywhere's messaging connectors. Other ALE-driven applications can define, subscribe to, and receive these XML reports through web services as well.

☞ For more information about RFID Anywhere's ALE implementation, see [“Using ALE to Filter and View RFID Data” on page 153](#).

## Data Protocol Processors

A **Data Protocol Processor**, or DPP, is used to encode and decode raw RFID tag data. DPPs can be created by developers to turn raw RFID tag data, including custom data, into valuable information with business context, making it easier for that data to be used by business logic. For example, a DPP could be created to extract the various fields encoded in the custom data portion of an ISO 18000-6B tag. If the custom data portion of the tag was used for internal IT asset tracking, it may contain information identifying the department name and employee number associated with the asset. No matter how this data is encoded, whether with encryption or according to some other rules, a DPP makes it very simple for developers to obtain each specific field to assist with business logic or processing. With RFID Anywhere's DPP capabilities, organizations have the utmost flexibility when choosing which tags to use, and how to encode data on them. Once created and installed into RFID Anywhere, DPPs can be used throughout the various components of RFID Anywhere.

☞ For more information about the benefits of using DPPs, see [“Getting Started with Data Protocol Processors \(DPPs\)” on page 95](#).

☞ For more information about creating a DPP, see [“Creating and Using a Data Protocol Processor \(DPP\)” \[RFID Anywhere Developer's Guide, page 65\]](#).

## Messaging connectors

RFID Anywhere's included **messaging connectors** abstract low-level aspects of messaging protocols such as TCP and SMTP and enable RFID Anywhere components to send processed information to enterprise systems or other components. RFID Anywhere includes a number of messaging connectors including File Messaging, MSMQ, QAnywhere, SMTP, TCP, UDP, and web services. These connectors can be added and configured from the Administrator Console. As with hardware connectors, developers do not need to understand the low-level intricacies of a given messaging system when developing a business module or configuring report delivery from the

Administrator Console. Developers need only to be concerned with generating and outputting the data in the correct format. The list of messaging connectors to send the data through can be specified when using the Administrator Console to define where reports or business module output should be sent. This enables business logic running in RFID Anywhere to easily communicate with a wide variety of enterprise applications.

☞ For more information about the messaging connectors available with RFID Anywhere, see [“RFID Anywhere Messaging Connectors” on page 101](#).

## Administrator Console

The RFID Anywhere **Administrator Console** is a web-based administration tool that you can use to access the components of your RFID Anywhere system.

You can perform the following tasks using the Administrator Console:

- ◆ start, edit, restart, stop, and remove instances of local services using the Services Manager
- ◆ view and set host system properties and settings for specific services using the Properties Manager
- ◆ access the monitoring and provisioning functions of RFID Anywhere
- ◆ add and manage managed systems in your RFID Anywhere network
- ◆ add new instances of service types
- ◆ edit local security and domain security

☞ For more information about the Administrator Console, see [“The Administrator Console” on page 31](#).

## RFID Anywhere Visual Studio Extension

For .NET developers, RFID Anywhere includes a Microsoft **Visual Studio .NET extension** to create custom business modules. These business modules can process data and integrate with existing enterprise applications. With this extension, RFID Anywhere automatically generates C# code containing all of the necessary environmental references, allowing the developer to focus on core business logic.

The framework is tightly integrated with Visual Studio .NET to provide a powerful and easy-to-use development environment. This integration allows a developer to use a set of predefined templates and wizards to speed up the

development process and to incorporate newly-created components with some standard functionality. Repetitive tasks such as registering for notifications or accessing specific hardware commands and features are automated by wizards.

☞ For information about using the RFID Anywhere Visual Studio Extension to create custom business modules, see [“Using the RFID Anywhere Visual Studio Extension to Create Custom Business Modules”](#) [*RFID Anywhere Developer’s Guide*, page 7].

## RFID Label Designer

The **Label Designer** is used to design the layout of RFID labels. It is a graphical tool that provides a number of controls, such as dynamic text fields and bar codes, that can be used to design an RFID label. The Label Designer generates XML documents containing the label design that are imported by specific RFID printer connectors. These designs are then translated by the connector into the format required by the specific printer, and are loaded into the printer’s memory so that the labels can be printed and encoded with raw RFID tag data.

☞ For more information about the Label Designer application, see [“The Label Designer”](#) on page 89.

## RFID Anywhere service discovery, loading, and communication

RFID Anywhere provides a dynamic management infrastructure. This advanced architecture allows RFID Anywhere to automatically discover all deployed components that need to be managed when the service first starts. Newly-deployed components will be detected automatically as well. This is different than other tools that offer management capabilities which need to be configured in advance to manage specific components. By automatically discovering what needs to be managed, RFID Anywhere enables a plug-and-play architecture that allows new components, such as new hardware connectors or custom business modules, to be deployed to the core service.

When the RFID Anywhere service is started, it automatically discovers the available services by examining attributes of assemblies in specific folders. Once the available services are discovered, RFID Anywhere calls the start or configure command on each service depending on the auto-start setting of the service. Service dependencies are implemented, and a catalog is created for dynamic management. Proxies are also created on the fly to reflect business module methods that have the [RfidNetWebMethod] attribute for exposing the methods as web services.

Since RFID Anywhere dynamically loads any connector, controller, or business module it finds, it is recommended that you only install the specific components you need for each RFID Anywhere node. This will conserve resources on the computer running RFID Anywhere.

A number of RFID Anywhere components create and run in their own application domains. Separate application domains are used for security purposes, and to ensure that failure in one component does not bring down the entire system. RFID Anywhere automatically provides the capability to publicly expose a URI for individual application domains to simplify the communication between components running on the same computer or different computers. .NET remoting is used because it is the fastest way to communicate between components at a low level. This enables RFID Anywhere to process data and work in terms of milliseconds.

RFID Anywhere loads the available services in the following order:

**1. Core business modules**

Business modules flagged as core business modules start before other components, so they cannot rely on other components such as controllers being started already. These components are part of the core RFID Anywhere system and can not be removed.

**2. Controllers**

Controllers start before regular custom business modules so that custom business modules can attach to them. Each controller is its own unique application domain, and each controller loads its specific connectors into its memory. Controllers on different RFID Anywhere nodes communicate with each other using .NET remoting to enable the aggregation of data and discovery of connectors regardless of location.

**3. Custom business modules**

Based on dependencies, RFID Anywhere loads custom business modules last. Each business module is also its own application domain.

## RFID Anywhere security overview

RFID readers can be located nearly anywhere: at a receiving dock, on a truck, or in a kiosk. Such environments are usually far from centralized servers and may have only occasional access to a wide area network or the Internet. RFID Anywhere was designed for installation at the same physical site as the readers and tags it services at the edge of the network. To reduce network traffic and total overhead costs, output from an RFID network must be processed early in its life cycle. Data must be captured and analyzed at the same physical location as the readers themselves. This ability to extract meaningful, valuable information on a real-time basis is called edge processing, and it is at the heart of RFID Anywhere.

Operating at the edge of the network exposes valuable corporate data to certain risks. RFID Anywhere provides the functionality to secure data from the edge to the enterprise. RFID Anywhere integrates tightly with local or directory-based authentication mechanisms. This protects access to the system, and enables role-based business logic execution and the role-based access to RFID network management consoles. Communication traffic is secured using web services security and certificates. This ensures that data is protected as it moves from the edge to the enterprise.

### Controlling access to your RFID network

In a distributed environment, it is important to have management capabilities that provide a complete view of the system, and that simplify common administrative tasks. RFID Anywhere provides browser-based management consoles to monitor and manage the RFID network. Enterprise Manager provides a single view of the RFID network. The Administrator Console provides management capabilities for a single node of the RFID network.

The management consoles expose a significant amount of information about your RFID network. This information can be exchanged between the Enterprise Manager and other nodes on the network or when users access the Administrator Console. Web services security (WS-S) ensures that information is securely exchanged to the browser and between nodes of the RFID Anywhere network. RFID Anywhere signs certificates, encrypts the information, and transmits it over secure HTTP. In addition, Microsoft Message Queuing (MSMQ), which is used to transport information between the RFID Anywhere nodes, controls who can send and receive this information.

Controlled access to the Administrator Console is ensured by requiring that users log in and authenticate themselves. User authentication can use local security or network security policies such as LDAP, Active Directory, or

Windows Domain. Local security authentication can be leveraged for integration with hardware and distributed environments. Local security can be managed from the Administrator Console enabling security at the edge without requiring local management at the edge.

Network security policy authentication allows RFID Anywhere to take advantage of existing corporate security implementations. It is also useful for enabling role-based security. In addition to requiring users to authenticate, role-based security ensures that the ability to perform actions depends on what rights are given. The advantage of role-based security is that access can be restricted to specific aspects of the RFID network without impeding other management requirements.

## Managing access to the business logic

RFID Anywhere nodes are responsible for the collection and edge processing of RFID and other collected data. As discussed above, custom business modules interact with hardware, query and update secondary sources such as databases and other applications, and send processed information to enterprise systems. With this broad capability, there is a need to manage access to that business logic. RFID Anywhere delivers with the ability to run role-based business logic. RFID Anywhere leverages its tight integration with local or directory-based authentication mechanisms to determine the user roles and send that information to the business module. The business module developer is responsible for implementing the desired role-based security. This allows greater flexibility to implement controlled access to the business logic.

## Installation instructions

The following steps are required to install and run RFID Anywhere 2.0.1 on your computer:

1. Install MSMQ Messaging Service (if it is not already installed).  
☞ For instructions, see [“Installing MSMQ” on page 16](#).
2. Install Microsoft Internet Information Services (IIS) (if it is not already installed).  
☞ For instructions, see [“Installing IIS” on page 17](#).
3. Install the Microsoft .NET Framework 1.1 SP1 and enable ASP.NET.  
☞ For instructions, see [“Installing the .NET Framework 1.1 SP1” on page 18](#).
4. Install Microsoft Web Services Extensions (WSE) 2.0 SP3 for Microsoft .NET (if it is not already installed).  
☞ For instructions, see [“Installing WSE 2.0 SP3” on page 19](#).
5. Install RFID Anywhere 2.x core files.  
☞ For instructions, see [“Installing RFID Anywhere 2.x core files” on page 20](#).
6. Install RFID Anywhere 2.0.1 updates.  
☞ For instructions, see [“Installing RFID Anywhere 2.0.1 updates” on page 21](#).
7. Install the RFID Anywhere Visual Studio Extension. (This is only required if you are using the Visual Studio Extension to develop custom business modules).  
☞ For more instructions, see [“Installing the RFID Anywhere Visual Studio Extension \(optional\)” on page 23](#).
8. Ensure that all required Windows services are running.  
☞ For instructions, see [Ensuring required services are running](#).
9. Start the RFID Anywhere service.  
☞ For instructions, see [“Starting the RFID Anywhere service” on page 25](#).

### System requirements

You need the following to use RFID Anywhere:

- ◆ 512 MB RAM

- ◆ 2 GHz processor
- ◆ Windows XP Professional SP2 or Windows Server 2003 SP1
- ◆ MSMQ messaging service
- ◆ Microsoft Internet Information Services
- ◆ .NET Framework 1.1 SP1, including ASP.NET
- ◆ Microsoft Web Services Extensions 2.0 SP3
- ◆ Microsoft Visual Studio .NET 2003 (only required if you want to install the RFID Anywhere Visual Studio Extension)

## Installing MSMQ

Before you can install and use RFID Anywhere, you must have Microsoft Message Queuing (MSMQ) installed on your computer. MSMQ is used throughout RFID Anywhere to enable components to communicate with one another.

### ❖ **To determine if MSMQ is already installed on your computer**

1. From the Start menu, choose Settings ► Control Panel.  
The Control Panel opens.
2. Double-click Add or Remove Programs.  
The Add or Remove Programs dialog opens.
3. On the left side of the Add or Remove Programs dialog, click Add/Remove Windows components.  
The Windows Components wizard opens.
4. Select Message Queuing in the Components list, and then click Details.
5. If the component titled Common is selected, the necessary MSMQ components are already installed on your computer. If the component titled Common is not selected, follow the instructions below to install MSMQ on your computer.

❖ **To install MSMQ on your computer**

1. From the Start menu, choose Settings ► Control Panel.  
The Control Panel opens.
2. Double-click Add or Remove Programs.  
The Add or Remove Programs dialog opens.
3. On the left side of the Add or Remove Programs dialog, click Add/Remove Windows components.  
The Windows Components wizard opens.
4. Select Message Queuing in the Components list, and then click Details.
5. Ensure the component titled Common is selected to be installed, and then click OK.
6. Follow the instructions in the wizard, insert your Windows CD if prompted, and click Finish when the wizard completes.

## Installing IIS

RFID Anywhere uses IIS to host the ASP.NET pages for the Administrator Console and to expose Web Services to other applications.

❖ **To determine if IIS is already installed on your computer**

1. From the Start menu, choose Settings ► Control Panel.  
The Control Panel opens.
2. Double-click Add or Remove Programs.  
The Add or Remove Programs dialog opens.
3. On the left side of the Add or Remove Programs dialog, click Add/Remove Windows components.  
The Windows Components wizard opens.
4. Select Internet Information Services (IIS), and then click Details.
5. If the components titled Common Files, Documentation, Internet Information Services Snap-in, and World Wide Web Service are selected, the necessary IIS components are already installed on your computer. Follow the instructions below to verify that IIS is functioning correctly. If the above components are not selected, follow the instructions below to install IIS on your computer.

❖ **To install IIS on your computer**

1. From the Start menu, choose Settings ► Control Panel.  
The Control Panel opens.
2. Double-click Add or Remove Programs.  
The Add or Remove Programs dialog opens.
3. On the left side of the Add or Remove Programs dialog, click Add/Remove Windows components.  
The Windows Components wizard opens.
4. Select Internet Information Services (IIS), and then click Details.
5. Ensure the components titled Common Files, Documentation, Internet Information Services Snap-in, and World Wide Web Service are selected to be installed, and then click OK.
6. Follow the instructions in the wizard, insert your Windows CD if prompted, and click Finish when the wizard completes.

❖ **To determine that IIS is functioning correctly**

1. Open the Control Panel by choosing Start ► Settings ► Control Panel.
2. Double-click Administrative Tools.
3. Double-click Services.
4. Ensure that the following services have a Status of Started, and that their Startup Type is set to Automatic:
  - ◆ IIS Admin
  - ◆ World Wide Web Publishing
5. From a web browser, navigate to <http://localhost>. An ‘Under Construction’ page or custom web site should appear.

☞ For more information on troubleshooting IIS issues, see the online resources at <http://www.microsoft.com>.

## Installing the .NET Framework 1.1 SP1

The Microsoft .NET Framework 1.1 SP1 is used to enable the ASP.NET pages of the Administrator Console and to provide additional tools and libraries for RFID Anywhere and custom business logic. If the Administrator Console does not display or function correctly after installation, it may indicate that this step did not complete successfully.

❖ **To determine if the .NET Framework 1.1 SP1 is installed on your computer**

1. The Microsoft article 318785 located at <http://support.microsoft.com/?kbid=318785&sd=msdn#XSLTH3166121122120121120120> outlines the recommended steps for determining if the .NET Framework 1.1 SP1 is installed on your computer.
2. If the .NET Framework 1.1 SP1 is not installed, follow the instructions below to install it. If it is installed, follow the step in the instructions below to ensure that ASP.NET for IIS is enabled, as this can not be verified using the Microsoft article above.

❖ **To install the .NET Framework 1.1 SP1 on your computer**

1. Download and execute the Microsoft .NET Framework version 1.1.4322 Redistributable Package from <http://www.microsoft.com/downloads/details.aspx?FamilyID=262d25e3-f589-4842-8157-034d1e7cf3a3&displaylang=en>.
2. From a command prompt, navigate to `C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\` and run the command `aspnet_regiis.exe -I` to ensure that ASP.NET for IIS is enabled.
3. Download and execute the Microsoft .NET Framework 1.1 Service Pack 1 package (KB867460) from <http://www.microsoft.com/downloads/details.aspx?FamilyID=a8f5654f-088e-40b2-bbdb-a83353618b38&displaylang=en>. Restart your computer if you are prompted.

## Installing WSE 2.0 SP3

Microsoft Web Services Extensions (WSE) 2.0 SP3 for Microsoft .NET is used to enable web services from RFID Anywhere.

❖ **To determine if WSE 2.0 SP3 is installed on your computer**

1. From the Start menu, choose Settings ► Control Panel.  
The Control Panel opens.
2. Double-click Add or Remove Programs.  
The Add or Remove Programs dialog opens.
3. Verify that Microsoft WSE 2.0 SP3 is in the list of currently installed programs. If it is, WSE 2.0 SP3 is installed on your computer. If it is not, follow the instructions below to install WSE 2.0 SP3 on your computer.

❖ **To install WSE 2.0 SP3 on your computer**

1. Download and execute the Microsoft Web Services Extensions 2.0 SP3 for Microsoft .NET package from <http://www.microsoft.com/downloads/details.aspx?familyid=1ba1f631-c3e7-420a-bc1e-ef18bab66122&displaylang=en>.
2. Click Next to proceed.
3. Read and accept the terms in the license agreement, and then click Next.
4. Select Runtime to install the runtime files, and then click Next.
5. Click Install to proceed with the installation.
6. Click Finish once the installation is complete. Close any dialogs that appear.

## About the RFID Anywhere installation program

The RFID Anywhere core installation program consists of a high-level Microsoft Installer (MSI) package that launches separate installers for each component to be installed. The MSI packages for components available to be installed are included in the *Packages* folder alongside the high-level install program. The Custom installation option allows the administrator to choose which components to install. If the Complete install option is chosen, all components with MSI files in the *Packages* folder are installed.

Updates to RFID Anywhere, such as the updates for version 2.0.1, are packaged as MSIs that must be deployed from the RFID Anywhere Administrator Console's Provisioning page.

**Note**

RFID Anywhere updates, in the form of MSI files, must be deployed through the RFID Anywhere Administrator Console's Provisioning page. Double-clicking these files to execute the MSI file will not correctly install the update.

## Installing RFID Anywhere 2.x core files

❖ **To install RFID Anywhere 2.x core files**

1. Download the RFID Anywhere installation program.
2. Extract the ZIP file you downloaded to your desktop or a temporary location.

3. In the directory where you extracted the RFID Anywhere installation, open the *RFIDAnywhereCore* folder.
4. Double-click *RFID Anywhere 2.0.msi*.  
The RFID Anywhere 2.0 Setup wizard opens. Click Next.
5. Select the country where you are installing the software. Click Next.
6. Choose whether you agree to the License Agreement.  
You must agree to the RFID Anywhere License Agreement before you can proceed with the installation.
7. Enter the Order Number, Feature Name, and Authorization Code from your RFID Anywhere license, and then click Next.
8. Enter your name and organization in the respective fields on the Customer Information dialog, and choose to install RFID Anywhere for all users of the computer, or only for yourself.
9. Select Component Manager, Group Manager, or Site Manager to specify the type of RFID Anywhere node, and then click Next.  
To review the different types of RFID Anywhere nodes, see [“A powerful distributed architecture” on page 5](#).
10. Select Complete to install all components of RFID Anywhere in the default directory of *C:\Program Files\Sybase\RFID Anywhere*. To customize the program features that are installed or to change the installation directory, select Custom. Click Next.
11. If you selected Custom, select the program features to install and specify the installation directory for RFID Anywhere. Click Next.
12. Click Install to begin the installation.
13. Click Close when the installation completes.  
RFID Anywhere is now installed on your computer.

## Installing RFID Anywhere 2.0.1 updates

RFID Anywhere updates are deployed using RFID Anywhere’s Provisioning page, which is also used to deploy hardware connectors.

To learn more about the RFID Anywhere Administrator Console and Provisioning, see [“The Administrator Console” on page 31](#) and [“The Provisioning page” on page 47](#)

❖ **To deploy and install the RFID Anywhere 2.0.1 updates**

1. Open the Administrator Console in Internet Explorer by browsing to **<http://localhost/rfidanywhere>**.

2. Log in with an account that has administrator credentials for your computer, such as a local user account in the Administrators group.

3. Click Provisioning.

The Provisioning page appears.

4. In the Deploy New Packages section, click Immediately in the Deployment Schedule list to deploy the package right away, or specify a date and time for the package should be deployed.

5. Click Browse in the Package field, and select the *SetupRFIDCore.2.0.145.msi* file from the *RFIDAnywhereUpdates201* directory where you extracted the RFID Anywhere installation. Click Open.

The selected file appears in the Package field.

6. Click Deploy.

A message appears at the bottom of the Deploy New Packages section of the page indicating that the package was deployed successfully.

The deployed package is listed in the Packages section of the page.

7. After the RFID Anywhere services restart, follow the above steps to deploy *SetupSimulator.2.0.151.msi* and *SetupFormDesigner.2.0.149.msi* which update the RFID Network Simulator and the RFID Label Designer respectively.

## Installing the RFID Anywhere help files

The RFID Anywhere documentation set is available online. The help files for default RFID Anywhere components such as core business modules and messaging connectors can be downloaded and extracted to your RFID Anywhere installation directory. This will make the help files available when configuring a connector or business module from the Administrator Console.

The RFID Anywhere help files, along with detailed instructions for downloading and installing, can be found at [http://www.ianywhere.com/developer/product\\_manuals/rfid\\_anywhere/index.html](http://www.ianywhere.com/developer/product_manuals/rfid_anywhere/index.html).

## Ensuring the required services are running

Before starting RFID Anywhere, it is important to ensure that other Windows services required by RFID Anywhere are running.

### ❖ To ensure the required Windows services are running

1. Open the Control Panel by choosing Start ► Settings ► Control Panel.
2. Double-click Administrative Tools.
3. Double-click Services.
4. Ensure that the following services have a Status of Started, and that their Startup Type is set to Automatic:
  - ◆ IIS Admin, used to manage the web site
  - ◆ Message Queuing, used internally by RFID Anywhere and by the MSMQ messaging connector
  - ◆ World Wide Web Publishing, used to host the Administrator Console and web services.

## Installing the RFID Anywhere Visual Studio Extension (optional)

This section explains how to install the RFID Anywhere Visual Studio Extension on your computer. The extension provides C# templates that you can use to write custom business modules for RFID Anywhere in Visual Studio .NET. Microsoft Visual Studio .NET 2003 must be installed on your computer before running this installation.

### ❖ To install the RFID Anywhere Visual Studio Extension

1. In the directory where you extracted the RFID Anywhere installation, open the *VSExt* folder.
2. Double-click *Setup.Exe*.

The RFID Anywhere (VS Extension) Setup wizard opens. Click Next.
3. Select the country where you are installing the software. Click Next.
4. You must agree to the RFID Anywhere License Agreement before you can proceed with the installation. Click Next.
5. Specify the installation directory for the RFID Anywhere Visual Studio Extension and specify which users can access the RFID Anywhere Visual Studio Extension. Click Next.

6. Click Next on the Confirm Installation page to start the installation.
7. Click Close to complete the installation.

The RFID Anywhere Visual Studio Extension is now installed on your computer.

☞ For information about using the RFID Anywhere Visual Studio Extension, see [“Using the RFID Anywhere Visual Studio Extension to Create Custom Business Modules”](#) [*RFID Anywhere Developer’s Guide*, page 7].

## Creating a local administrator account

To access the RFID Anywhere Administrator Console, authentication credentials must be entered. RFID Anywhere is configured to allow users from the computer’s Administrators group and newly-created RFIDManagers group to access the Administrator Console. It is useful to create a local user account with Administrator privileges when doing development and administering an RFID Anywhere node, as members of the Administrators group have additional management capabilities.

### ❖ To create a new local administrator account

1. From the Start menu, choose Settings ► Control Panel.

The Control Panel opens.

2. Double-click Administrative Tools.
3. Double-click Computer Management.
4. Expand Local Users and Groups.

5. Right-click Users and choose New User from the popup menu.

The New User dialog appears.

6. Enter the User Name, the Full Name, and Description for the new account.

7. Enter the Password, and then clear the user must change password at next logon option.

8. Click Create to create the account.

9. Click Close to return to the Users folder.

10. Right-click the newly-created account and choose Properties.

The Account property sheet appears.

11. On the Member Of tab, click Add.  
The Select Groups dialog appears.
12. Click Advanced to refine your search.
13. Click Find Now to search all of the groups on the computer.
14. Select Administrators, and then click OK.
15. Click OK again to return to the Properties window.
16. Click Apply to apply the changes, and then click OK to return to the Groups folder.
17. Close Computer Management.

## Starting the RFID Anywhere service

Before you start using RFID Anywhere, you must ensure the RFID Anywhere service is running. The RFID Anywhere service is created with a Startup Type of Automatic, so the service starts after a new installation completes. You can change this service to have a manual startup, which requires the service to be started before it can be used.

### ❖ To start the RFID Anywhere service

1. From the Start menu, choose Programs ► RFID Anywhere ► Start Service.

The RFID Anywhere Service starts running on the computer. This service is named `rnservice` and launches the process `iAnywhere.RfidNet.RunTime.exe`, which is visible from the Windows Task Manager.

## Post-install analysis

This section provides a brief summary of the components included in your RFID Anywhere installation, including the directory structure.

### RFID Anywhere shortcuts

By default, the RFID Anywhere installation places an RFID Anywhere program group on the Start menu under Programs. This program group contains shortcuts for accessing the Administrator Console, Documentation, RFID Label Designer, and RFID Simulator Data Editor, as well as menu items for starting and stopping the RFID Anywhere service.

- ◆ **Administrator Console** This shortcut opens the RFID Anywhere Administrator Console. The shortcut launches the default web browser and opens the Administrator Console Login page (<http://localhost/RFIDAnywhere/>).
- ◆ **Documentation** This shortcut opens the RFID Anywhere documentation. When clicked for the first time, the RFID Anywhere documentation page opens in your default web browser. The *RFID Anywhere Getting Started Guide*, *RFID Anywhere Developer's Guide*, and *Administrator Console help files* are available to download from this page. When new versions of the documentation are released, they are posted on this web site.
- ◆ **RFID Label Designer** This shortcut is used to access the RFID Label Designer. The RFID Label Designer is a graphical tool that provides a number of controls, such as dynamic text fields and bar codes, that can be used to build an RFID label.
- ◆ **RFID Simulator Data Editor** This shortcut is used to access the RFID Simulator Data Editor. The RFID Simulator Data Editor is a graphical tool that allows developers to test their business logic by creating RFID tags and simulating an RFID environment.
- ◆ **Start Service and Stop Service** The Start Service and Stop Service shortcuts start and stop the RFID Anywhere service, respectively.

### IIS virtual directory

The RFID Anywhere installation will create an Internet Information Service (IIS) virtual directory named *RFIDAnywhere*. It is mapped to the local RFID Anywhere root directory.

### ❖ To access RFIDAnywhere virtual directory

1. From the Start menu, choose Settings ► Control Panel.
2. Double-click Administrative Tools.
3. Double-click Internet Information Services.
4. In the left pane, select Default Web Site under Internet Information Services ► Local Computer ► Web Sites.

The RFID Anywhere virtual directory appears in the right pane.

## Digital certificate

An RFID Anywhere digital certificate is installed, which can be used to communicate securely with an RFID Anywhere business module via web services.

## Directory structure

The RFID Anywhere installation copies many directories and files to the RFID Anywhere root directory. By default, the location of the root directory is `C:\Program Files\Sybase\RFID Anywhere`.

Root directory	Stores the web services, Administrator Console pages, executables for engine and editor, configuration files, and log files.
bin	Contains the core RFID Anywhere framework used for developing business modules, core business modules, and any installed device connector(s).
bms	Contains custom business modules. When a custom business module is deployed or built, a [GUID] folder is created. The business module DLL files are stored under their respective GUID folders.
cache	RFID Anywhere uses this folder internally to store information in memory during run time.
data	Contains <i>rn</i> files of all connectors and business modules that are deployed in the Administrator Console Local Services list. The <i>rn</i> files store property values that are modified by the user from the Properties Manager. The files are separated according to the service GUIDs.
Deploy	Contains the RFID Anywhere engine that starts the run-time and creates application domains.
docs	Once the files are downloaded from the RFID Anywhere documentation web site, this folder contains all of the documentation files.
FormDesigner	Contains the RFID Label Designer executable and all the required libraries.

log	Contains the logs (.mlg files) for all connectors and business modules that are started from the Administrator Console. For each service, one file is generated per day and they are separated according to the service GUIDs. The folder also contains files that record system logs and errors.
packages	Contains packages that are stored for deployment at a later time. These packages can be installed via Provisioning.
PublicKey	Contains the RFID Anywhere security certificate.
WS	Contains helper .aspx files for use by RFID Anywhere

## RFID Anywhere configuration file

The RFID Anywhere runtime configuration file (*iAnywhere.RfidNet.RunTime.exe.config*) is located in the RFID Anywhere root directory. It allows you to configure the port on which to listen for managed system setup, for controllers to communicate with each other in a distributed environment and for .NET remoting. The file is in XML format and can be modified using a generic text editor. It is important to note that although the configuration file provides access to some server properties, the majority of the configuration should be performed via the Administrator Console.

## Uninstalling RFID Anywhere

To uninstall RFID Anywhere and the RFID Anywhere Visual Studio Extension, use the Add or Remove Programs functionality from the Control Panel. This will remove all connectors, business modules and other packages installed by RFID Anywhere.

### ❖ To uninstall RFID Anywhere

1. Go to Start ► Programs ► RFID Anywhere ► Stop Service
2. Once the RFID Anywhere service stops, go to Control Panel ► Add or Remove Programs
3. Select any MSI files that you may have installed after RFID Anywhere was installed. For example, select InventoryTrackerSetup (if you have it installed). Click Remove.

The selected file(s) will be uninstalled and removed.

4. Select RFID Anywhere Visual Studio Extension (if you have it installed) and click Remove.

The RFID Anywhere Visual Studio Extension will be uninstalled and removed.

5. Select RFID Anywhere and click Remove.

The core RFID Anywhere files and any MSI file(s) installed with the RFID Anywhere installation or through Provisioning, such as InventoryTrackerFiles or hardware connectors, will be uninstalled and removed.



---

## CHAPTER 2

# The Administrator Console

About this chapter

This chapter explains how to use the Administrator Console to administer your RFID Anywhere network.

Contents

<b>Topic:</b>	<b>page</b>
Using the Administrator Console	32
The Login page	33
The Home page	34
RFID Anywhere licensing	42
Monitoring an RFID Anywhere network	43
The Provisioning page	47
The Configuration page	51
Editing local security	53
Editing domain security	55

## Using the Administrator Console

The **Administrator Console** is a web-based administration tool that you can use to access the components of your RFID Anywhere system.

You can perform the following tasks using the Administrator Console:

- ◆ start, edit, restart, stop, and remove instances of local services using the Services Manager
- ◆ view and set host system properties and settings for specific services using the Properties Manager
- ◆ access the monitoring and provisioning functions of RFID Anywhere
- ◆ add and manage managed systems in your RFID Anywhere network
- ◆ add new instances of service types
- ◆ edit local security and domain security

**Note**

Internet Explorer is required to access the Administrator Console.

❖ **To start the Administrator Console from the RFID Anywhere computer**

1. Start the RFID Anywhere Service. From the Start menu, choose Programs ► RFID Anywhere ► Start Service.

The RFID Anywhere service starts running on your computer.

2. From the Start menu, choose Programs ► RFID Anywhere ► Administrator Console.

The Administrator Console Login page opens.

Alternatively, you can open Internet Explorer and type

**http://localhost/RfidAnywhere** in the Address Bar to open the Administrator Console.

The RFID Anywhere Login page appears.

❖ **To start the Administrator Console from another computer**

1. Open Internet Explorer and enter **http://server.company.com/RfidAnywhere**, where *server.company.com* represents the network address of the RFID Anywhere computer.

## The Login page

The Login page of the RFID Anywhere Administrator Console ensures that only authorized users can gain access to RFID Anywhere's administration capabilities. Members of the computer's Administrators group are permitted to log in to the Administrator Console. In addition, users in the RfidManagers group are also permitted to log in to the Administrator Console.

### ❖ To log in to the Administrator Console using a local account

1. In the User Name field, enter a valid local Administrator user that exists on the system. This account should be either a local user in the computers Administrators group, or a member of the RfidManagers group.
2. In the Password field, enter the user's password. Blank passwords are not permitted.
3. Select the local computer name in the Domain field.
4. Click Log On.

The Home page opens. If an error occurs while authenticating the user, an error message appears below the Log On button to assist in troubleshooting.

### ❖ To log in to the Administrator Console using a domain account

1. In the User Name field, enter the user name of a valid domain account that is part of the Administrators or RfidManagers group.
2. In the Password field, enter the user's password. Blank passwords are not permitted.
3. In the Domain field, select the name of the domain. Domains include the corporate domain that the RFID Anywhere computer is a part of, and a server directory if configured.

☞ For more information on configuring authentications against a server directory such as Microsoft Active Directory or Novell Directory, see [“Editing domain security” on page 55](#).

4. Click Log On.

The Home page opens. If an error occurs while authenticating the user, an error message appears below the Log On button to assist in troubleshooting.

## The Home page

The Administrator Console Home page includes the Services Manager, the Properties Manager, and Managed Systems.

**Services Manager** The Services Manager displays instances of all the local services available on your RFID Anywhere network.

Local services include connectors (hardware or messaging) and business modules (custom or core) installed on the RFID Anywhere node. Every single instance of a hardware connector, messaging connector, or business module is represented by a unique instance of a service.

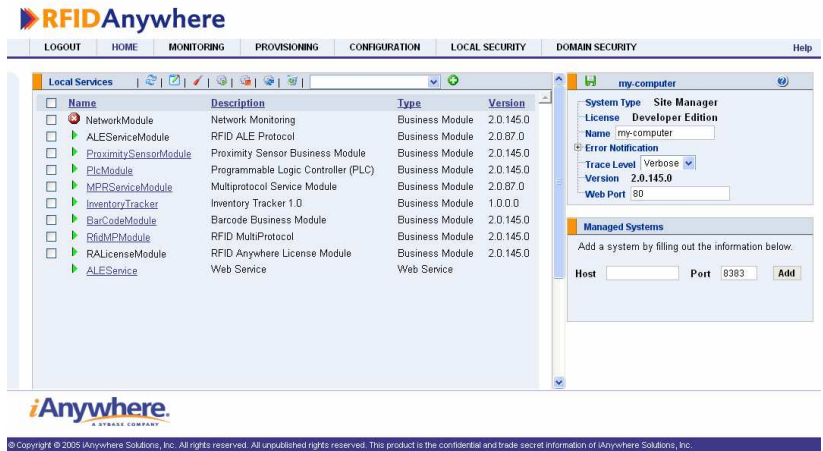
**Properties Manager** The Properties Manager displays the property settings for a service that is selected for editing in the Services Manager. If no services are being edited, the Properties Manager displays the properties for the RFID Anywhere system.

**Managed Systems** The Managed Systems pane allows you to add or remove **managed systems** from your RFID Anywhere network. These additional Administrator Consoles are managed by the **Site Manager** (the computer where RFID Anywhere is installed).

## Managing local services

**Local services** refer to instances of service types that have been added during RFID Anywhere installation, using the Service Type dropdown list, or installed from the Provisioning page. Local services include connectors (hardware or messaging) and business modules (system or custom) installed on the RFID Anywhere node. Every single instance of a hardware connector, messaging connector, or business module is represented by a unique instance of a service. Each of the services listed is associated with a specific set of properties that appear in the Properties Manager.

All local services are managed using the **Services Manager**. Click Home in the Administrator Console to view a list of services in your RFID Anywhere network.



You can use the Services Manager to perform the following tasks:

- ◆ View or edit the properties of a service using the Properties Manager.
- ◆ View or edit business module properties created in Visual Studio .NET using the Properties Manager.
- ◆ View or edit instances of service types added from the Service Type dropdown list.
- ◆ Edit, configure, start, stop, restart or remove one or more local services.
- ◆ Refresh the entire Local Services listing.

Information about the Local Services installed by RFID Anywhere is included throughout this manual.

## Adding local services

Adding a new local service adds an instance of a service as a connector. Connectors enable you to communicate with hardware devices such as RFID readers in your RFID Anywhere network and configure communication pathways to your enterprise applications.

Custom business modules are added to the list of local services in the Administrator Console automatically after they are compiled and deployed.

☞ For instructions on deploying custom business modules to your RFID network, see [“Deploying packages” on page 47](#).

❖ **To add a new local service type**

1. In the Administrator Console, choose a local service type from the Service Type dropdown list.
2. Click Add, which is represented by a green plus sign icon beside the Service Type dropdown list.  
The new local service appears in the Local Services List.
3. Start or edit the local service as required.

## Starting local services

Once you have added a local service to your network, you must start it before you can use it to communicate with hardware or enterprise applications in your RFID Anywhere network.

❖ **To start a local service**

1. In the Administrator Console, select a local service from the list of services in the Name column.
2. Click Start in the Local Services menu, which is represented by a gear icon with a green arrow.  
A green arrow appears beside the service name indicating that the service is running. A red X next to a service name indicates that the service is stopped.

## Editing local service properties

You can edit the properties of local services in the Administrator Console.

❖ **To edit the properties of a local service**

1. Select a service from the list of services in the Name column.
2. In the Local Services menu, click Edit, which is represented by a window and pencil icon.  
The Properties Manager displays the properties values for the corresponding service.
3. Use the text fields and dropdown lists to view or modify the properties of the service. Use the + sign to expand nested properties, or properties that can contain multiple values.

4. After modifying property values, click the green disk in the Properties Manager title bar to save any changes.

Click the blue question mark in the Properties Manager or refer to the documentation for more information on specific property values and their meanings.

## Refreshing the Local Services list

Refreshing the Local Services list updates the contents of the Local Services list in the Administrator Console.

### ❖ To refresh the Local Services list

1. In the Local Services menu, click Refresh, which is represented by two green arrows.

The Local Services list displays the most recent local service information.

## Stopping local services

You can stop local services using the Administrator Console. Once you stop a local service, it cannot be accessed by other components in your RFID Anywhere network. Stopping a service does not remove the item from the RFID Anywhere network. You can restart a stopped service at any time.

### ❖ To stop a local service

1. Select a service from the Local Services list.
2. In the Local Services menu, click Stop, which is represented by a gear icon with a red box.

A red X appears beside the service name, indicating that the service is stopped.

## Restarting local services

You can restart stopped local services using the Administrator Console.

### ❖ To restart a local service

1. Select a stopped service in the Local Services list.
2. In the Local Services menu, click Restart, which is represented by a gear icon with two blue arrows.

A green arrow appears beside the service, indicating that it is started.

## Removing local services

You can remove local services using the Administrator Console. Once you remove a service from your RFID Anywhere network, it cannot be restarted and it cannot be accessed by other components in your RFID Anywhere network.

### ❖ To remove a local service

1. Select a local service from the list of services in the Name column.
2. In the Local Services menu, click Remove, which is represented by a garbage can icon.  
A confirmation dialog appears.
3. Click OK.

The service is removed from the list and from the RFID Anywhere network.

## Configuring local services

You can configure local services using the Administrator Console. Clicking the Configure link when a connector is selected will call a connector's Configure method if one is implemented. For example, the Configure method of a printer connector pre-loads forms into memory.

### ❖ To configure a local service

1. Select a local service from the list of services in the Name column.
2. In the Local Services menu, click Configure, which is represented by a screwdriver icon.  
The service is configured.

## Using the Properties Manager

The Properties Manager displays property settings related to a local service when a property is being edited. When no services are being edited, or when the Home page is first loaded, the Properties Manager displays the properties for the local computer running RFID Anywhere.

Properties that have values you can edit have displayed textboxes or dropdown lists in the Properties Manager. Read-only properties are displayed in bold text. When a property of a connector or a business module is modified from the Properties Manager, a *.rn* file is created. The *.rn* file

stores the values of properties exposed in the Properties Manager and is located in the `data/[GUID]` folder of your RFID Anywhere root directory.

The editable fields and dropdown lists used to set property values allow you to control how RFID Anywhere displays information, reports error conditions for troubleshooting, and communicates with your enterprise system.

## Using the Properties Manager help

The blue question mark that appears beside the service name at the top of the Properties Manager opens the help for that service. Click the blue question mark to view descriptions of the properties listed in the Properties Manager. The Properties Manager help contains detailed descriptions of each of the property fields included for the service that is selected in the Local Services list.

### ❖ To view the Properties Manager help

1. Select a local service in the Services Manager list and then click Edit.
2. Click the blue question mark in the Properties Manager.

The Properties Manager help appears in a new browser window.

When you write new business modules and add them to the Services Manager, you have the option of providing custom help content to describe the properties that can be configured in the Properties Manager.

☞ For information about adding help for your custom business modules, see [“Supplying help for business modules in the Administrator Console”](#) [*RFID Anywhere Developer’s Guide*, page 103].

## Managed Systems

The Managed Systems pane allows you to add RFID Anywhere nodes to your RFID Anywhere network. Having multiple nodes in your RFID Anywhere network allows you to load balance system processing for optimal efficiency, and to perform RFID simulations from more than one location. RFID Anywhere nodes that are managed by the Site Manager or Group Manager are called **managed systems** of that specific node.

Each console that you add is assigned a specific host name and port number. RFID Anywhere allows you to create a flexible network as you can add as many managed systems to the network as you require. This provides the ability to build a hierarchical network of many RFID Anywhere nodes. The parent node is automatically fed data from the child nodes, as the controllers

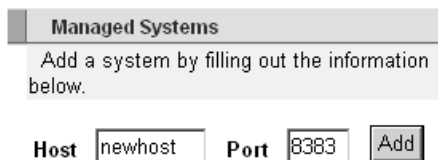
on each system communicate with each other at a very low level. For example, business modules running in the higher tiers will be able to subscribe to and get data from hardware connectors running on lower tiers. Error notifications move up the network hierarchy in the same manner. Connectors in a distributed architecture must have unique names. It is recommended that the unique computer name be used as a prefix to the connector name to ensure uniqueness.

## Adding managed systems

You can have multiple RFID Anywhere nodes in your RFID Anywhere network. Additional nodes are managed by the Administrator Console on the Site Manager or Group Manager node.

### ❖ To add a managed system

1. Open the Administrator Console on the Site Manager or Group Manager computer.
2. In the Managed Systems pane, type the IP address or host name for the host in the Host field and the port number for the host in the Port field.  
Port 8383 is the default management port. This port is configured in the **groupport** key from the node's *iAnywhere.RfidNet.RunTime.exe.config* configuration file.



Managed Systems

Add a system by filling out the information below.

Host  Port

3. Click Add.  
The new host system appears in the Managed Systems pane.

## Accessing managed systems

You can view any of the managed systems in your RFID Anywhere network using the Administrator Console on a Site Manager or Group Manager.

### ❖ To view managed systems

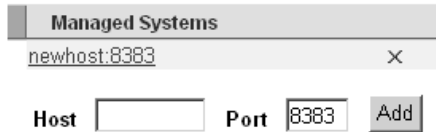
1. Open the Administrator Console on the Site Manager or Group Manager.
2. Click a host name in the Managed Systems list.  
The selected host opens in a separate browser window.

## Removing managed systems

You can only remove managed systems from the Administrator Console on the Site Manager or Group Manager.

### ❖ To remove a managed system

1. In the Managed Systems pane, click the X beside the name of the managed system you want to remove.



The system is removed from the Managed Systems list and from your RFID Anywhere network.

## RFID Anywhere licensing

RFID Anywhere is licensed in one of three modes depending on the required usage.

The evaluation edition of RFID Anywhere is licensed for 60 days for evaluation purposes only. This edition of the software does not enable hardware support, with the RFID Network Simulator being an important part of the evaluation process.

The developer edition of RFID Anywhere provides non-expiring use of RFID Anywhere for development purposes only. This license enables hardware connectors to be installed for use with RFID Anywhere. This edition is used when hardware connectivity is required, or when usage longer than 60 days is required.

Deployment licenses for RFID Anywhere are required to deploy applications developed with RFID Anywhere.

👉 To obtain a developer edition license or to purchase deployment licenses, visit [www.ianywhere.com/RFID](http://www.ianywhere.com/RFID).

### Updating a license key

To install RFID Anywhere, you must have a license key for one of the editions listed above. If you have obtained a new license key after installation, either to move from evaluation to development, development to deployment, or simply to extend an evaluation, you must update the license key from the Administrator Console.

#### ❖ To update a license key

1. Browse to the Home page of the Administrator Console.
2. From the Services Manager, select the checkbox beside the RFID Anywhere License Module, RALicenseModule, and click Edit.
3. In the text box for the Key property, enter the updated license key.
4. Click the green disk in the Property Manager title bar to save the changes.  
The Edition, License, LicenseKey, and License Status properties are updated to reflect the new license.

## Monitoring an RFID Anywhere network

**Monitoring** allows you to view system log file information and log file events for all of the local services installed on your RFID Anywhere enterprise system. It also allows you to identify and troubleshoot system operations issues. All monitoring tasks are performed on the Monitoring page of the Administrator Console.

When a service is added to the Administrator Console, a log file is created automatically. The Monitoring page of the Administrator Console allows you to view log information filtered by date, message type, and other options.

The system monitoring and events information listed on the Monitoring page comes from the local services, enterprise host system, and managed systems that run your RFID Anywhere network from the Administrator Console.

System-level log files are stored in the log folder, which is located in the root directory of your RFID Anywhere installation and have a *.mlg* extension. A new log file is created every day, and the date is appended to the log file name, for example, *RfidMPModule 10-19-2005.mlg*. The information stored in the log files can be viewed on the Monitoring page of the Administrator Console and is filtered using the System Monitoring and Events lists. These files are saved in XML format, so they can also be viewed with other viewing or test editing tools.

### System Monitoring list

The System Monitoring list includes the names, descriptions, and types of services running on your RFID Anywhere system. You can view information about the following service types:

- ◆ **System** shows information about local services related to the host computer(s) running RFID Anywhere on your RFID Anywhere network.
- ◆ **Connectors** shows information about the connectors that communicate with each physical device associated with RFID Anywhere. One connector is required for each type of physical device.
- ◆ **Controllers** shows information about events that run internally on the RFID Anywhere system.
- ◆ **Business modules** shows information about custom-coded *.msi* software packages in your RFID Anywhere network that are designed to work with RFID Anywhere on your enterprise system.

### Events list

The Events list displays a list of the events that have occurred for each service by time, type, and message. The events are associated with the service selected from the System Monitoring section of the page. Log file

information is created daily and can be viewed by date using the Date dropdown list.

You can add subscribers to the errors listed on the Monitoring page using any of the supported messaging connectors.

☞ For information about adding subscribers to event notifications, see [“Adding messaging connectors to error notifications” on page 45](#).

☞ For information about the different messaging connectors available, see [“RFID Anywhere Messaging Connectors” on page 101](#).

## Obtaining system information

The Monitoring page of the Administrator Console contains information about log files for local services.

### ❖ To monitor system information

1. Click Monitoring in the Administrator Console.  
The Monitoring page appears.
2. Choose a service from the System Monitoring dropdown list.  
Log file information for the selected service appears in the Events section of the page.
3. Use the Date dropdown list to view information stored from previous dates if required.

## Refreshing the Events list

Refreshing an event updates the log information associated with the selected log file in the RFID Anywhere network.

### ❖ To refresh an event

1. Select a service in the System Monitoring dropdown list.  
The events associated with the log file appear in the Events section of the page.
2. Click Refresh in the Events menu.  
All logs associated with the selected log file are updated.

## Removing events from the Events list

Removing event information in the Administrator Console removes the

specified information from the error log files. You cannot retrieve event information once it has been removed.

❖ **To remove events**

1. Open the Administrator Console and then click Monitoring.
2. In the System Monitoring dropdown list, select the item whose events you want to remove.
3. Select the date of the events you want to remove using the Events dropdown list.
4. Click Remove.

The event information is removed. Events associated with the service are removed from RFID Anywhere.

## Clearing an event

Clearing an event deletes the contents of the log file associated with the selected service and deletes any log files that are older than the specified date.

❖ **To clear events**

1. Open the Administrator Console and then click Monitoring.
2. In the System Monitoring dropdown list, select the item whose events you want to remove.
3. Select the date of the events you want to clear using the Events dropdown list.
4. Click Clear in the Events menu.

The contents of the log file for the specified date are cleared. Any log files that are older than the specified date are deleted.

## Adding messaging connectors to error notifications

You can receive notification about RFID Anywhere errors using the messaging connectors provided with RFID Anywhere. For example, a configured SMTP messaging connector can be used to send email alerts to administrators when errors occur, or a configured TCP connector can be used to sent error notifications to a third-party monitoring tool.

❖ **To add messaging connectors for system error notifications**

1. Open the Administrator Console.
2. Click Home to open the system properties in the Properties Manager page.
3. Expand the Error Notification property, and enter the name of the first messaging connector you want to use for error notifications in the text field and click the green plus sign to add the messaging connector to the list.
4. To add additional messaging connectors, enter their names one at a time in the text box, pressing the green plus sign to add them to the list, and pressing the red X to remove them.
5. Click the green disk in the Properties Manager title bar to save your changes.

❖ **To add messaging connectors for business module error notifications**

1. Open the Administrator Console.
2. Select the business module you want to add messaging connectors to send error notifications about, and then click Edit.
3. Expand the Error Notification property, and enter the name of the first messaging connector you want to use for error notifications in the text field and click the green plus sign to add the messaging connector to the list.
4. To add additional messaging connectors, enter their names one at a time in the text box, pressing the green plus sign to add them to the list, and pressing the red X to remove them.
5. Click the green disk in the Properties Manager title bar to save your changes.

## The Provisioning page

About the Provisioning page

**Provisioning** allows you to install, deploy, uninstall, or remove packages created for use with RFID Anywhere. All provisioning tasks are performed on the Provisioning page of the Administrator Console.

A package is an *.msi* file that contains either a business module created by a developer, or a connector from iAnywhere. When deployed, the packages appear in RFID Anywhere as local services in the Administrator Console.

Packages are deployed using the Deploy New Packages section of the Provisioning page. Deployed packages are automatically versioned to provide the ability roll back to previous settings should operational issues arise with a newly-deployed package.

☞ For more information, see [“Package versioning” on page 48](#).

When packages are deployed, their contents are automatically copied to the appropriate directory where they can be discovered by RFID Anywhere. Business modules are installed to a unique folder in the *bms* folder of your RFID Anywhere installation, and connectors are deployed to the *bin* folder. Once deployed, the RFID Anywhere service is automatically restarted so the new components are detected.

## Deploying packages

When you deploy a package, it is installed automatically on your RFID Anywhere network. You can deploy *.msi* files you have created in Visual Studio .NET to add your own custom business modules to your RFID Anywhere network.

☞ For information about creating MSI deployment packages for custom business modules, see [“Deploying business modules” \[RFID Anywhere Developer’s Guide, page 25\]](#).

### ❖ To deploy and install a package

1. Open the Administrator Console.
2. Click Provisioning.  
The Provisioning page appears.
3. In the Deploy New Packages section, click Immediately in the Deployment Schedule list to deploy the package right away, or specify a date and time for the package should be deployed.
4. Click Browse in the Package field, and select the package file in the Choose File dialog box. Click Open.

The selected file appears in the Package field.

5. Click Deploy.

A message appears at the bottom of the Deploy New Packages section of the page indicating that the package was deployed successfully.

The deployed package is listed in the Packages section of the page.

The Status column of the Packages section indicates the status of the deployed package. Status options are:

- ◆ **Queued** the package is queued to be installed or uninstalled.
- ◆ **Installed** the package has been installed successfully and is available for use.
- ◆ **Failed** the package installation failed.

You can click Refresh in the Packages menu to update the Status column.

**Note**

The RFID Anywhere service automatically restarts when a package is deployed. You must allow the service time to restart before clicking Refresh. An error message may appear if the service is restarting when you click Refresh.

## Package versioning

RFID Anywhere automatically performs version control on packages you deploy from the Provisioning page. Each time a package is deployed, RFID Anywhere looks at the ProductCode property of the MSI package, which is specified by the developer when the package is created, to determine the package version. The version number appears in the Version column of the Packages section of the page. You can keep as many old versions of a package as necessary. If you want to install an older version of a package, you must first uninstall any newer versions of the package.

When a new version of a package is deployed, RFID Anywhere archives the previous version. The status of an archived package is Stored. You can use the stored package to revert to the previous version should operational issues arise with the use of a newly-deployed package.

## Scheduling packages for deployment

Scheduling a package deployment for deployment allows you to choose the optimal time to deploy the package.

❖ **To schedule a package for deployment**

1. Start the Administrator Console.
2. Click Provisioning.  
The Provisioning page appears.
3. Select Specific Date and Time to deploy the package for a date in the future. Type a specific date in the Date field, or click the ellipses (...) to select a date using the calendar tool. Specify a deployment time by choosing the hour, minute, and time of day (A.M. or P.M.).
4. Click Browse in the Package field, and locate the package file using the Choose File dialog.
5. Click Open.  
The selected file appears in the Package field.
6. Click Deploy.  
A message appears at the bottom of the Deploy New Packages section of the page indicating the scheduled deployment date and time. Once the package is successfully installed, it appears in the Administrator Console in the Local Services list.

## Refreshing packages

Refreshing the Packages list on the Provisioning page allows you to view the most current status of all the packages installed or stored on your RFID Anywhere system.

❖ **To refresh the Packages list**

1. Click Refresh in the Packages menu.  
The list is updated to reflect the most recent information for all of the packages in the list.

## Uninstalling packages

Uninstalling a package from the Provisioning page stores the associated *.msi* file in an archive folder in your RFID Anywhere system. The stored package can be reinstalled at any time by clicking Install in the Packages menu.

❖ **To uninstall a package**

1. Select a package in the Packages list.
2. Click Uninstall in the Packages menu.  
The package is removed from the list.
3. Click Refresh in the Packages menu.

The status of the package changes to Stored. You can now click Remove to completely remove the package.

If you are uninstalling your own custom business module MSI package, the MSI package is stored in the *Packages\Repository* folder of your RFID Anywhere installation directory. This gives you easy access to the package if you want to reinstall it at a later date.

**Note**

The RFID Anywhere service automatically restarts when a package is uninstalled. You must allow the service time to restart before clicking Refresh. An error message may appear if the service is restarting when you click Refresh.

## Removing packages

Removing a package from the Provisioning page deletes the associated *.msi* file from the system. You can not retrieve or reinstall a package once it has been removed.

❖ **To remove a package**

1. Select a package in the Packages list.
2. Click Remove in the Packages menu.  
A confirmation dialog appears.
3. Click OK to continue.  
The package is removed from the list and from your RFID Anywhere network.
4. Click Refresh in the Packages menu to update the list.

## The Configuration page

The Configuration page of the Administrator Console exposes RFID Anywhere system properties as they relate to the RFID Anywhere service and the web site. Although some of these properties can also be configured from the Internet Services Manager, exposing them from the web-based Administrator Console allows system administrators to update these properties from any location without requiring them to access the operating system, which is extremely important in a distributed network.

Since the Configuration page exposes management of computer and IIS settings, only users that have logged in to the Administrator Console who are part of the computer's Administrators group are able to navigate to the Configuration page. For users that are part of the RfidManagers group, but not the Administrators group, the link to the Configuration page is greyed out.

The System section of the Configuration page exposes service-level properties of RFID Anywhere. The RFID Anywhere service can be started, stopped, or restarted from the title bar of the System section.

The System section exposes the following properties:

- ◆ **System Type** This read-only property identifies the RFID Anywhere instance as either a Site Manager, Group Manager, or Component Manager, which was determined at installation time.
- ◆ **Maximum Number of Threads** This property identifies the maximum number of threads that RFID Anywhere will create and use per process. The default value is 300.
- ◆ **Recycle Schedule** This optional property instructs RFID Anywhere to shut down, reload, and restart all services at a specified day and time. The format of this property is *day:hour :minute*, where *day* is a number between 0 and 7 (0=every day, 1=Monday, 2=Tuesday, 3=Wednesday, 4=Thursday, 5=Friday, 6=Saturday, 7=Sunday), *hour* is a number between 00 and 23, inclusive (24-hour clock), and *minute* is a number between 00 and 59 inclusive.

A value of 0:23:30 would recycle the RFID Anywhere service every day at 11:30 P.M.

If you are specifying single digit integers for hour or minute, you must add a 0 when entering the values. For example, to recycle the RFID Anywhere service every Monday at 1:01 A.M., use 1:01:01, not 1:1:1.

- ◆ **Heartbeat Time (in minutes)** This property tells the RFID Anywhere node how often to send a heartbeat message to Enterprise Manager if an

Enterprise Manager connection is specified. The default value is 10.

- ◆ **Network Connection** This property specifies the connection to use when sending heartbeat messages to Enterprise Manager. For example, if a dial-up configuration is used, the Site Manager will dial up, connect, send catalogs, and then terminate the dial-up connection.
- ◆ **IP Address** If a specific network connection with an IP address is specified in the Network Connection property, the IP address of the system appears in this property.

The IIS Settings section of the Configuration page exposes IIS properties and functionality that can be managed from the Administrator Console. The IIS service can be started, stopped, or restarted from the title bar of the IIS Settings section.

The IIS Settings section exposes the following properties:

- ◆ **Port Number** This property identifies the network port that IIS (and ultimately the RFID Anywhere Administrator Console and Web Services) is listening on.
- ◆ **SSL Port Number** If SSL is enabled, this property identifies the network port to be listened on for SSL connections.
- ◆ **Connection Timeout** This property identifies a time value, in seconds, after which an idle connection to the server times out.
- ◆ **Certificate Name** If SSL is enabled, this property identifies the SSL certificate being used by the server. This property lists the certificates available to be used. RFID Anywhere installs a personal certificate named RFID Anywhere that can be used to enable SSL.

## Editing local security

The Local Security page of the Administrator Console exposes local user and group management so the administrator can create local users and update user membership in the **RfidManagers** and **RfidWebServiceUsers** groups. Users in the RfidManagers group can log in the Administrator Console. Users in the RfidWebServiceUsers group are permitted to call Web Service methods running in RFID Anywhere. Although some of the user and group management settings can also be configured from the Local Users and Groups section of the operating system's Computer Management interface, exposing them from the web-based Administrator Console allows system administrators to update these properties from any location without requiring them to access the operating system, which is extremely important in a distributed network.

Since the Local Security page exposes management of user accounts on the local computer, only users that have logged in to the Administrator Console who are part of the computer's Administrators group are able to navigate to the Local Security page. For users that are part of the RfidManagers group, but not the Administrators group, the link to the Local Security page is greyed out.

The top section of the Local Security page lists the local users currently configured on the computer. The users are displayed in groups of 5, with the arrows used to navigate forward or backward in the list of users. Each user has an associated checkbox in the RfidManagers and RfidWebServiceUsers column, with a selected checkbox indicating membership in the respective group. The Refresh link is used to refresh the displayed list of users.

### ❖ To change a user's group membership

1. Use the arrows to browse to the user to be added to the group.
2. Click the pencil beside the user to edit the user's group membership.
3. Select or clear the checkbox in the appropriate group column.
4. Click the green checkmark to save group membership changes.

### ❖ To delete a user

1. Use the arrows to browse to the user to be added to the group.
2. Click the delete icon in the delete column beside the user.

The bottom section of the Local Security page allows an administrator to create new local users who can be added to the RfidManagers or RfidWebServiceUsers group.

❖ **To create a new local user**

1. Enter a unique user name in the Username field.
2. Optionally, type a description for the new user in the Description field.
3. Type a password for the new user in the Password field.
4. Retype the password to confirm the Confirm Password field.
5. Click Create User to create the new user.

## Editing domain security

The Domain Security page of the Administrator Console allows administrators to configure user authentication against an LDAP-based server directory such as a Microsoft Active Directory or Novell Directory.

Since the Domain Security page exposes management of user accounts and connections to enterprise servers, only users that have logged in to the Administrator Console who are part of the computer's Administrators group can access the Domain Security page. For users that are part of the RfidManagers group, but not the Administrators group, the link to the Domain Security page is greyed out.

The top section of the Domain Security page is used to configure the connection between RFID Anywhere and the server directory. To change the connection details, click the pencil beside Directory Authentication. The User Directory Type dropdown list allows the administrator to choose the type of server directory being used. The IP address or DNS name of the server directory is entered in the host field. The server directory port, often port 389 for LDAP-based directories, is entered in the port field. If the server directory supports SSL communication, the Enable SSL option can be selected, with the SSL port being entered in the Port field. The Username and Password fields are used to specify the user credentials that are to be used to access the server directory. Once updates are made to these fields, click the green checkmark to save the changes. This instructs RFID Anywhere to attempt to establish a connection to the server directory and to query the available groups.

The bottom section of the Domain Security page allows administrators to define how to map server groups to the RfidManagers and RfidWebServiceUsers groups. To change the group mappings, click the pencil beside Domain/Group Settings. The Domain/Naming Context field is used to specify the distinguished name format of the server. If the Directory Authentication connection settings were entered correctly above, and if a connection to the directory server was established correctly, the dropdown list beside the Domain/Naming Context field will be populated with available options. Selecting an option in this list populates the Domain/Naming Context field. Then, the RFID Managers Group field is used to specify the server directory group to use to define users that have permission to log in to the RFID Anywhere Administrator Console. The RFID Web Service Users Group field is used to specify the server directory group to use to define users that have permission to invoke web services running in RFID Anywhere. Once updates are made to these fields, click the green checkmark beside Domain/Group Settings to save the changes. This completes the configuration of communication to a server directory for

RFID Anywhere authentication, and adds this directory to the list of Domains on the Login page of the RFID Anywhere Administrator Console.

To test RFID Anywhere's domain security features, it is necessary to have access to an existing server directory, or to a test environment. Tools such as Microsoft's ADAM (Active Directory Application Mode), available at <http://www.microsoft.com/downloads/details.aspx?FamilyId=9688F8B9-1034-4EF6-A3E5-2A2A57B5C8E4&displaylang=en>, allow users to install a simple directory server to test the domain security capabilities of products and applications like RFID Anywhere.

---

## CHAPTER 3

# Default RFID Anywhere Services

About this chapter

This chapter introduces the RFID Anywhere services that are installed by default.

Contents

**Topic:**

**page**

---

[Default local services](#)

58

## Default local services

RFID Anywhere installs several local services by default. You can view these services in the Local Services list in the Administrator Console Home page. This chapter describes each of the default services and provides a list of supported properties. A number of other services introduced here are only installed and visible if they have been installed.

### MPRServiceModule (Multiprotocol Service Module)

The MPRServiceModule is a business module that is an essential component of RFID Anywhere's Report Engine MP development model. It is used to specify groups of readers that the RFID MultiProtocol Report Definition connectors can recognize.

Grouping readers under a common name can be very useful. Using the same example as the one provided for RFID ALE Protocol, suppose there are four readers in a warehouse: two located at the shipping/receiving doors, and two located near the shelves. Using the Multiprotocol Service Module, you can group the readers so that the two readers at the doors are grouped under **doors** and the two readers near the shelves are grouped under **shelves**. In your RFID MultiProtocol Report Definition connectors, you need only specify the group names and do not have to go through the trouble of entering each individual reader connector. Grouping also makes your solution future-proof. Suppose the warehouse decides to add three additional readers near the shelves. Then you would only need to update the Multiprotocol Service Module to include the new readers. You would not need to update the RFID MultiProtocol Report Definition connectors since they only care about group names.

☞ For information on RFID Anywhere's Report Engine MP development model or the RFID MultiProtocol Report Definition connector, see [“Using Report Engine MP” on page 183](#).

The following table lists the properties of the Multiprotocol Service Module that are accessible via the Properties Manager.

Property	Description
Description	A description of the business module. This property is not configurable. The default value is Multiprotocol Service Module.

Property	Description
Name	The name of the business module. This property is not configurable. The default value is MPRServiceModule.
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>
Logical Readers	<p>The relationship between the group names and reader connectors are specified in this property.</p> <p>You must enter each statement on a separate line, for example:</p> <pre>doors:reader1\antennal doors:reader2\antennal shelves:reader3\antennal shelves:reader4\antennal</pre> <p>You can also enter RFID MP Simulator connectors. For example:</p> <pre>doors:RfidMPSimulator1</pre>

## RfidMPModule (RFID Multiprotocol)

RFIDMPModule is a business module that exposes RFID Multiprotocol controller methods via web services. It can be used to send tag information when a tag event occurs. Tag events occur when tags appear or disappear from the read range of an RFID reader. The RFID reader(s) to read tag information from and the messaging connector(s) to transmit data to external

applications can be specified via Properties Manager in the Administrator Console Home page.

The following table lists the properties of the RFID Multiprotocol business module that are accessible via the Properties Manager.

Property	Description
Description	A description of the business module. This property is not configurable. The default value is RFID Multiprotocol.
Name	The name of the business module. This property is not configurable. The default value is RfidMPModule.
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
OnTagEvent	The names of the messaging connector(s) used to send tag information. When a tag event occurs, the information on the tags is sent via the messaging connectors listed in this property. By adding configured messaging connectors to this property, reader events can be sent to enterprise applications without any development. For example, you could enter the names of any messaging connectors that appear in the Local Services list. The event output messages sent to messaging connectors for delivery are simple XML documents.
Version	The version of the service.

Property	Description
Sources	<p>The names of RFID reader connectors that are configured to detect tags. The RFID Multiprotocol business module obtains tags and tag event notifications from these readers. You can include the names of RFID MP Simulator connectors.</p>
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

The RFID Multiprotocol business module also exposes various methods to Web Services. To obtain the Web Service Definition Language (WSDL) file

for the RFID Multiprotocol business module, click the RfidMPModule link under the Local Services list.

☞ For information on the available methods, see “[Multiprotocol RFID reader controller](#)” [*RFID Anywhere Developer’s Guide*, page 36].

## ALEServiceModule (RFID ALE Protocol)

The RFID ALE Protocol business module is an essential component of RFID Anywhere’s ALE development model. It is used to specify groups of readers that the ALE Event Cycle connector can recognize.

Grouping readers under a common name can be very useful. For example, suppose there are four readers in a warehouse: two located at the shipping/receiving doors, and two located near the shelves. Using the RFID ALE Protocol business module, you can group the readers so that the two readers at the doors are grouped under **doors** and the two reader near the shelves are grouped under **shelves**. Then, in your ALE Event Cycle connectors, you need only specify the group names and do not have to go through the trouble of entering each individual reader connectors. Grouping also makes your solution future-proof. Suppose the warehouse decides to add three additional readers near the shelves. Then you only need to update the RFID Ale Protocol business module to include the new readers. You would not have to update the ALE Event Cycle connectors since they only care about group names.

☞ For information on RFID Anywhere’s ALE development model or the ALE Event Cycle connector, see “[Using ALE to Filter and View RFID Data](#)” on page 153.

The following table lists the properties of the RFID ALE Protocol business module that are accessible via the Properties Manager.

Property	Description
Description	A description of the business module. This property is not configurable. The default value is RFID ALE Protocol.
Name	The name of the business module. This property is not configurable. The default value is ALEServiceModule.

Property	Description
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
Version	The version of the service.
Start Mode	A value indicating how the service starts. <ul style="list-style-type: none"><li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li><li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li></ul>
Synchronous Event Cycle Timeout	A value indicating the timeout value (in milliseconds) for event cycles running in a synchronous mode communicating with another application. The default value is 15000.

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>
Logical Readers	<p>The relationship between the group names and reader connectors are specified in this property.</p> <p>You must enter each statement on a separate line, for example:</p> <pre>doors:reader1\antennal doors:reader2\antennal shelves:reader3\antennal shelves:reader4\antennal</pre> <p>You can also enter RFID MP Simulator connectors. For example:</p> <pre>doors:RfidMPSimulator1</pre>

## ALEService (Web Service)

The ALEService is a business module that exposes ALE methods over Web Services. It provides a secondary means (first being the ALE Event Cycle connector) of interacting with the ALE engine by exposing methods over the web using SOAP standard.

Since some ALE applications may need reports only at specific times or only as a result of user action, developers can use the Web Service module to define and trigger one-time ALE reports or retrieve important parameters. To view the details of the methods available through RFID Anywhere’s ALE Service in the form of a WSDL file, click the **ALEService** link in the list of Local Services list from the Home page of the Administrator Console.

The ALE Service business module does not expose any of its properties in the Properties Manager pane. Thus it is not configurable from the Administrator Console.

☞ For information on the available ALE methods, see “[Available ALE methods](#)” [*RFID Anywhere Developer’s Guide*, page 54].

## NetworkModule (Network Monitoring)

The Network Module is used to scan a network segment to enable hardware connector discovery by MAC address. This is imperative for networks where hardware devices are connected through DHCP leases where the actual IP address can change and is not known at the time the hardware connector is configured. For any hardware connector configured with a MAC Address property and not an IP Address, the Network Module is able to discover the device’s IP address and configure the connector.

The following table lists the properties of the Network Monitoring module that are accessible via the Properties Manager pane.

Property	Description
Broadcasting	The default setting is True.
Broadcasts	The default value is an empty string.
Broadcast Timeout	By default, this value is set to 5 minutes.
Description	Description of the service. This property is not configurable. The default value is Network Monitoring.
Lost Timeout	The default value is 20 seconds.
Name	The name of the service. This property is not configurable. The default value is NetworkModule.

Property	Description
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
Ping	By default, this is set to True.
Ping Local Segment	By default, this is set to True.
Ping Ranges	IP ranges to ping to locate RFID hardware by MAC address. This property is set automatically when RFID Anywhere is started. If the IP segment of the RFID Anywhere computer changes, ensure this property gets updated.
Ping Timeout	The default value is 10 seconds.
Resolve Timeout	The default value is 10 seconds.
Version	The version of the service.
SNMP	By default, this is set to True.
SNMP Retries	The default value is 3.
SNMP Timeout	The default value is 2 seconds.
Socket Timeout	The default value is 1 second.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list. This is the default setting.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console.</li> </ul>

Property	Description
Trace Level	<p>Specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul> <p>Default value: Error</p>

## RALicenseModule (RFID Anywhere License Module)

The RALicenseModule is used to display information about the currently licensed edition of RFID Anywhere, and to provide a mechanism for updating the RFID Anywhere license key.

The following table lists the properties of the RFID Anywhere License module that are accessible via the Properties Manager pane.

Property	Description
Description	This property is not configurable. The default value is RFID Anywhere License Module.
Edition	This property identifies the specific edition (Evaluation, Developer, or Deployment) enabled by the current license key. This property is not configurable.
License	Descriptive text outlining the specific edition of RFID Anywhere enabled by the current license key. This property is not configurable.

Property	Description
LicenseKey	The current license key. To update your license key, edit and save this property.
LicenseStatus	Specifies whether the current license key is valid or invalid.
Name	The name of the service. This property is not configurable. The default value is RALicenseModule.
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
OrderNumber	The order number specified with the license information. To update a license key to change the order number, edit and save this property.
Product Install Date	The date when RFID Anywhere was installed. This property is not configurable.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list. This is the default setting.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console.</li> </ul>

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

☞ For more information on RFID Anywhere licensing, see [“RFID Anywhere licensing” on page 42](#).

## PrinterModule (Printer Business Module)

The PrinterModule business module exposes the functionality of the RFID printer controller through web services. This business module is installed alongside the RFID printer controller, and is only available if both components have been installed.

☞ For more information on the PrinterModule and the RFID printer controller, see [“RFID printer controller” \[RFID Anywhere Developer’s Guide, page 42\]](#).

The following table lists the properties of the business module that are accessible via the Properties Manager pane.

Property	Description
Description	A description of the service. This property is not configurable. The default value is Printer Business Module.

Property	Description
Name	The name of the business module. This property is not configurable. The default value is PrinterModule.
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
Version	The version of the service.
Start Mode	A value indicating how the service starts. <ul style="list-style-type: none"><li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list. This is the default setting.</li><li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console.</li></ul>

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## ProximitySensorModule (Proximity Sensor Business Module)

The ProximitySensorModule business module exposes the functionality of the proximity sensor controller through web services. This business module is installed alongside the proximity sensor controller and is only available if both components have been installed.

☞ For more information on the ProximitySensorModule and the proximity sensor controller, see [“Proximity sensor controller” \[RFID Anywhere Developer’s Guide, page 49\]](#).

The following table lists the properties of the business module that are accessible via the Properties Manager pane.

Property	Description
Description	A description of the business module. This property is not configurable. The default value is Proximity Sensor Business Module.
Name	The name of the business module. This property is not configurable. The default value is ProximitySensorModule.

Property	Description
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
OnSensor	Specifies the messaging connectors to send all Proximity Sensor events to. By adding configured messaging connectors to this property, proximity events can be sent to enterprise applications without any development. The event output messages sent to messaging connectors for delivery are simple XML documents.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list. This is the default setting.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console.</li> </ul>

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## PlcModule (Programmable Logic Controller (PLC))

The PlcModule business module exposes the functionality of the PLC controller through web services. This business module is installed alongside the PLC controller and is only available if both components have been installed.

☞ For more information on the PlcModule and the PLC controller, see “PLC controller” [*RFID Anywhere Developer’s Guide*, page 50].

The following table lists the properties of the business module that are accessible via the Properties Manager pane.

Property	Description
Description	A description of the business module. This property is not configurable. The default value is Programmable Logic Controller (PLC).
Name	The name of the business module. This property is not configurable. The default value is PlcModule.

Property	Description
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
OnPlcEvent	Specifies the messaging connectors to send all PLC events to. By adding configured messaging connectors to this property, PLC events can be sent to enterprise applications without any development. The event output messages sent to messaging connectors for delivery are simple XML documents.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list. This is the default setting.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console.</li> </ul>

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## BarCodeModule (Barcode Business Module)

The BarCodeModule business module exposes the functionality of the bar code reader controller through web services. This business module is installed alongside the bar code reader controller and is only available if both components have been installed.

☞ For more information on the BarCodeModule and the bar code reader controller, see “[Bar code scanner controller](#)” [*RFID Anywhere Developer’s Guide*, page 45].

The following table lists the properties of the business module that are accessible via the Properties Manager pane.

Property	Description
Description	A description of the business module. This property is not configurable. The default value is Barcode Business Module.
Name	The name of the business module. This property is not configurable. The default value is BarCodeModule.

Property	Description
Error Notification	The names of the messaging connector(s) to send messages. Depending on the setting of the Trace Level property, the appropriate messages generated in the business module are transmitted through the specified messaging connectors.
New Scan Event	Specifies the messaging connectors to send all bar code scan events to. By adding configured messaging connectors to this property, bar code scan events can be sent to enterprise applications without any development. The event output messages sent to messaging connectors for delivery are simple XML documents.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list. This is the default setting.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console.</li> </ul>

Property	Description
Trace Level	<p data-bbox="790 248 1123 361">This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul data-bbox="790 383 1153 743" style="list-style-type: none"><li data-bbox="790 383 1153 439">◆ <b>Off</b> No messages are written to the log file.</li><li data-bbox="790 453 1153 534">◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li><li data-bbox="790 548 1153 604">◆ <b>Warning</b> Only warning messages are written to the log file.</li><li data-bbox="790 618 1153 673">◆ <b>Info</b> Only informational messages are written to the log file.</li><li data-bbox="790 687 1153 743">◆ <b>Verbose</b> All messages are written to the log file.</li></ul>

---

## CHAPTER 4

# The RFID Simulator Data Editor

### About this chapter

This chapter describes how to use the RFID Simulator Data Editor to create sample RFID data that you can use to test your RFID Anywhere network.

This chapter provides procedures for simulating EPC tags. Similar procedures apply for other tags with valid Data Protocol Processors.

### Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Using the RFID Simulator Data Editor</a>	80
<a href="#">Working with simulator files</a>	82
<a href="#">Summary</a>	87

## Using the RFID Simulator Data Editor

When developing an RFID application, performing load testing or testing special cases can be difficult. You can use the RFID Simulator Data Editor to design sample RFID data for testing applications, without acquiring real RFID hardware. The data created with this tool can simulate the RFID tag data for tags returned by an RFID reader at specific time intervals. You can also create **tag groups** to simulate multiple readers or antennas. A tag group defines the encoding protocol type, RFID tag range, total quantity, and life cycle for a series of RFID tags.

### Starting the RFID Simulator Data Editor

The RFID Simulator Data Editor helps you design sample RFID data that you can use to test your RFID applications.

#### ❖ To start the RFID Simulator Data Editor

1. From the Start menu, choose Programs ► RFID Anywhere ► RFID Simulator Data Editor.

The RFID Simulator Data Editor opens.

2. From the Session menu, choose New.
3. From the Groups menu, choose New ► EPC.

A window opens. This is where the tags are created and configured.

The RFID Simulator Data Editor consists of several panes that contain information about the simulated tags for your RFID Anywhere network.

#### Tag Groups list

Located in the left side of the screen, the Tag Groups list provides summary information about the tag groups you have created for a session, and includes information about the start and life time for each tag group.

#### Group pane

The Group pane is used to specify the following information about a tag group:

**Name** The name of the tag group. The name must be an alphanumeric value.

**Quantity** The number of tags in the tag group. This value must be greater than zero.

#### Creation Mode pane

The Creation Mode pane contains the following options:

**Enter Tags Manually** This option allows you to create tags manually. To do this, users can either use the Encoding pane to create a number of tags at

once and have the RFID Simulator Data Editor do the encoding, or users can manually paste encoded tags into the Tags pane. This mode is useful for creating simulation files with non-sequential tag data.

**Create Tags Using Template** This option allows you to create tags automatically using a template. The RFID Simulator Data Editor then only saves the template definition which is interpreted by the simulator connector. This mode is useful for creating simple scenarios where tags are sequential, such as in a conveyor belt scenario.

**Increment Tags Per Session** This option automatically increments the incrementable fields of tags for each new session.

Encoding pane

The Encoding pane is used to provide the following information about a tag group:

**Type** The coding type for the tag group's tag type. For example, if you selected the EPC tag type, then the coding type might be EPC\_GIAI64.

**Property** A list of the properties associated with the selected tag type.

**Value** A numerical value within the displayed range for a specific property.

**Increment** An option that is available for incrementing the value of specific properties.

**Range** A predefined theoretical range for the property. Some properties, such as the CompanyPrefixIndex property of 64-bit encodings, have a smaller allowed range based on the EPC Global ONS tables at <http://www.onsepc.com/ManagerTranslation.xml>.

**Note**

The allowed ranges for the entries are continuously reviewed and updated automatically. Therefore, the theoretical ranges provided in the Range column may not accurately reflect the actual allowed range. The RFID Anywhere Simulator Data Editor checks for updates to the allowed ranges and modifies the code internally the first time tags created after the program is started. An error message will be displayed if the value entered is out of the current allowed range.

To update the allowed range, the RFID Simulator Data Editor makes an HTTP request to <http://www.onsepc.com/ManagerTranslation.xml>. If this request fails or is blocked, the allowed range will not be updated.

Tags pane

The Tags pane displays the tags that have been created for the current session. The tags are displayed in encoded hexadecimal format.

## Working with simulator files

The RFID Simulator Data Editor uses simulator (.smt) files to store the tag groups you create. A simulator file can contain one, or multiple tag groups.

For detailed instructions on how to use the RFID Simulator Data Editor to create a simulation file for a specific scenario, see [“Creating Simulated RFID Data” on page 137](#).

### ❖ To create a new EPC simulator file

1. Start the RFID Simulator Data Editor by choosing Start ► Programs ► RFID Anywhere ► RFID Simulator Data Editor.

2. From the Session menu, choose New.

A new session is created.

3. From the Groups menu, choose New ► EPC.

A new window opens where you can define your EPC tags.

4. Set the properties for the tag group and then from the Groups menu, choose Add.

The Session Items dialog appears.

5. Set the Start Interval and Life Time for the group.

The Start Interval field identifies the delay that will be used before the group of tags appears. Entering **0:0** indicates that this group of tags appears immediately when the simulation starts, and immediately after a preceding group.

The Life Time field identifies the amount of time the group of tags will remain in the simulated reader’s read zone. Entering **0:10** indicates that this group of tags will remain in the simulated reader’s read zone for 10 seconds. After 10 seconds, the next group of tags in the sequence, configured in the following steps, will appear.

6. Choose the appropriate DPP and class to generate from the DPP and Class to Generate drop-down lists, respectively. Select the RFID Anywhere default DPP to use the default RFID Anywhere DPP for EPC tags, and iAnywhere.RfidNet.Rfid.Tags.RFIDTagEPC\_Class0 to generate Class 0 tags. Click OK.

The tag group is added to your file.

7. From the Session menu, choose Properties.

The Session Properties dialog appears.

8. Add a description and specify the number of repetitions for the tag group, and then click OK.
9. Choose the appropriate DPP from the DPP dropdown list. Select the RFID Anywhere default DPP to use the default RFID Anywhere DPP for EPC tags.
10. Click Create Tags in the main window.  
The new tags appear in the Tags list.

❖ **To save a simulator file**

1. From the Session menu, choose Save As.  
The Save RFID Simulator Data As dialog opens.
2. Type a name for the simulator file and then click Save.

❖ **To open a simulator file**

1. Start the RFID Simulator Data Editor.
2. From the Session menu, choose Open.
3. Browse to the location of the simulator file you want to open.
4. Select the simulator file you want to open, and then click Open.  
The simulator file opens in the RFID Simulator Data Editor.

## Adding tag groups

The simulator files you create in the RFID Simulator Data Editor contain information about tag groups that represent tags in an RFID Anywhere network. Each tag group includes information about the encoding protocol type, RFID tag range, total quantity, and life cycle for a series of RFID tags. You can add as many tag groups as necessary to a simulator file.

❖ **To add a new tag group to a simulator file**

1. Open the RFID Anywhere Simulator Data Editor by choosing Start ► Programs ► RFID Anywhere ► RFID Anywhere Simulator Data Editor.
2. From the Session menu, choose New.
3. From the Groups menu, choose New ► EPC.

A window opens where you can define your EPC tags.

Note that additional options are available if additional DPPs have been installed to RFID Anywhere.

4. In the Group pane, type a name for the tag group and enter the number of tags in the group.

5. In the Creation Mode pane, specify how the tags will be created, either manually or using a template.

Optionally, select Increment Tags Per Session to automatically increment tags for each new session. When this option is selected, the serial number of each tag is incremented every time the simulation cycle runs.

6. In the Encoding pane, choose the encoding type from the Type dropdown list and then specify the properties for the tags.

7. From the Session menu, choose Properties.

The Session Properties window opens. Provide a description of the tag group and the number of repetitions. Click OK.

The value specified in the Repetitions field defines the number of times the combined tag groups are repeated. For example, if you create GroupA with a Life of 1 minute and GroupB with a Life of 2 minutes, then the whole cycle repeats after 3 minutes. The cycle would be GroupA, GroupB, GroupA, GroupB, and so on based on the specified number of repetitions. -1 specifies infinite repetitions, otherwise, the number of repetitions must be 1 or greater.

8. From the Groups menu, choose Add.

The Session Items dialog appears.

9. Specify the Start Time and Life Interval for the group of tags and then click OK.

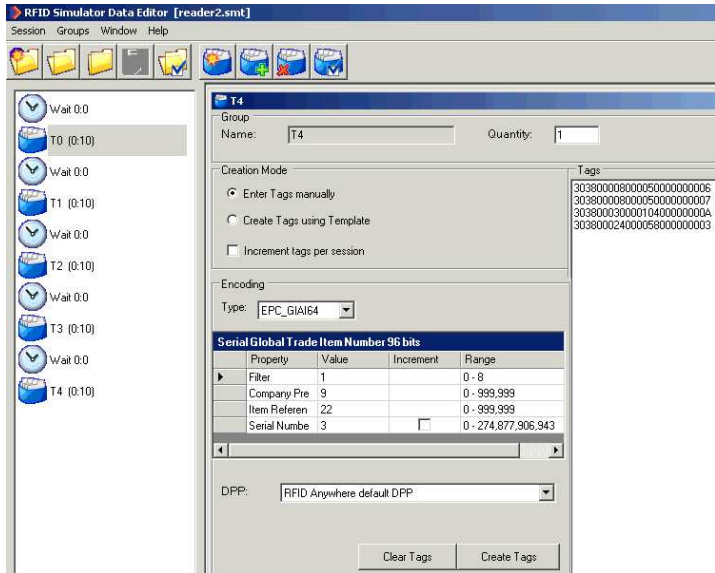
Choose the appropriate DPP and Class to generate from the DPP and Class to generate dropdown lists, respectively. Click OK.

The tag group is added to the Tag Groups list.

10. In the main window, select the appropriate DPP from the DPP dropdown list. You can choose the RFID Anywhere default DPP to use the default RFID Anywhere DPP.

11. Click Create Tags.

The new tags appear in the Tags list.



12. Save the simulator file.

☞ For more information about adding tag groups and using the RFID Data Simulator to test an RFID Anywhere network, see [“Creating Simulated RFID Data”](#) on page 137.

## Deleting tag groups

You can delete tag groups that you have added to a simulator file using the RFID Simulator Data Editor.

### ❖ To delete a tag group

1. Open a simulator file in the RFID Simulator Data Editor.
2. Select the tag group you want to delete in the Tag Groups list.
3. From the Groups menu, choose Delete.

A confirmation dialog appears.

4. Click OK.

The tag group is removed from the Tag Groups list.

## Configuring tag groups

Once you have created tag groups, you can edit the properties of the tags in the RFID Simulator Data Editor.

❖ **To configure a tag group**

1. Open a simulator file in the RFID Simulator Data Editor.
2. Click the tag group whose properties you want to edit in the Tag Groups list.
3. Configure the tag properties as required.

You can change the start interval and life time for a tag group. Select the tag group in the Tag Groups list and Under the Groups menu, and then choose Properties.

## Summary

This section introduced the RFID Simulator Data Editor. The tutorial [“Creating Simulated RFID Data” on page 137](#) takes you through the steps of creating a complete tag simulation based on a real-life scenario.



---

## CHAPTER 5

# The Label Designer

About this chapter

This chapter describes how to use the Label Designer application to design RFID labels that are used by RFID printer connectors to print RFID labels.

Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Using the Label Designer</a>	90
<a href="#">How label designs are used in RFID Anywhere</a>	93

## Using the Label Designer

The Label Designer is used to design the layout of RFID labels. It is a graphical tool that provides a number of controls, such as dynamic text fields and bar codes, that can be used to build an RFID label. The Label Designer generates XML documents containing the label design which are imported by specific RFID printer connectors. These designs are then translated by the connector into the format required by the specific printer, and are loaded into the printer's memory so that the labels can be printed and encoded with raw RFID tag data.

### Starting the Label Designer

The Label Designer is used as a standalone tool to define the components of an RFID label to enable printing.

#### ❖ To start the Label Designer

1. From the Start menu, choose Programs ► RFID Anywhere ► RFID Label Designer.

The Label Designer opens and a new Label Design is initiated.

The Label Designer consists of several panes that contain tools and information to enable the design of RFID labels.

#### Windows Forms list

Located on the left side of the screen, the Windows Forms list contains the available controls or items that can be included in a label design. Available controls include text fields, and a number of bar code formats.

#### Label Canvas pane

The Label Canvas pane, in the middle of the Label Designer screen, starts with a blank label canvas. Controls that result in printed items, such as text fields and bar codes, from the Windows Forms list can be added to and placed on the label canvas as desired. The label canvas can be resized as necessary, but it is important to note that the label canvas may not provide a completely accurate preview of a printed tag. The label canvas provides a guideline only, so it is recommended that you test label designs with the actual printer before finalizing them. Controls that do not reflect printed items, such as the RFID Tag control, are added to the bottom of the Label Canvas pane for configuration.

#### Properties pane

The Properties pane, on the right side of the Label Designer screen, outlines the specific properties of the layout or control that is currently selected. Properties can be easily modified using through the updating of text fields or dropdown lists as appropriate.

## Creating label designs

The Label Designer creates output files in XML format to store the label designs you create.

### ❖ To create a new label design

1. Start the Label Designer by choosing Start ► Programs ► RFID Anywhere ► RFID Label Designer.
2. From the File menu, choose New.  
A new label canvas is created.
3. Drag and place controls from the Windows Form pane onto the label canvas as desired. Controls can be deleted by selecting the control with your mouse and choosing Delete from the Edit menu.

To configure a control, select it with the mouse and update the properties in the Properties pane.

For dynamic controls, such as bar codes, dynamic text, and RFIDTag controls, make note of the VariableReference property and DynamicFieldValue property, as they are used by your business logic to populate the values. For example, if a DynamicText control is added to your label with a VariableReference property of **TXTDYNC** and a DynamicFieldValue property of **1**, business logic will refer to this control as **TXTDYNC1**, the concatenated value of those two properties.

For bar code controls, the Human Readable property indicates whether or not to print the numeric representation of the bar code on the label.

4. To give the label a unique name that allows it to be referenced later, click the header of the layout and specify the Text property, located in the Appearance set of properties.

### ❖ To print a test label

1. Add and configure an RFID printer control, such as the Printronix printer control. This entails configuring the IP address and port of the RFID printer.
2. From the Print menu, choose Label.

This uses the label design you have created to generate printer-specific commands that are sent to the printer. After examining the printed test label, modify the label design as appropriate.

❖ **To save a label design**

1. From the File menu, choose Save As.  
The Save As dialog opens.
2. Type a name and browse to a location for the label design XML file and then click Save.

❖ **To open a label design**

1. Start the Label Designer.
2. From the File menu, choose Open.
3. Browse to the location of the label design file you want to open.
4. Select the label design file you want to open, and then click Open.  
The label design file opens in the Label Designer.

## How label designs are used in RFID Anywhere

Once an RFID label has been designed, tested, and saved, RFID printer connectors are configured to import these files, convert them into the native printer language, and load them into the printer's memory. When the connector starts, the labels, which have already been converted into the native printer format, are loaded into the printer's memory. Once loaded into memory, these labels can be given values for dynamic fields (if required) and the printing process can be initiated from a custom business module.

☞ For more information about interacting with RFID printers and label designs from custom business modules, see “RFID printer controller” [*RFID Anywhere Developer's Guide*, page 42].

### Pre-loading label designs using a connector

RFID printer connectors installed in RFID Anywhere have configurable properties for importing label design files. The Label List property defines the number of labels that will be imported, converted, and pre-loaded into the printer's memory, up to a maximum of 10. Additional labels can be loaded by business modules.

For each label, two properties must be set:

- ◆ **Label Name** Gives a unique name for the label, and provides a way to reference the label from a custom business module
- ◆ **LabelFileName** Gives the complete path to the saved label design file



---

## CHAPTER 6

# Getting Started with Data Protocol Processors (DPPs)

About this chapter

This chapter introduces the RFID Anywhere Data Protocol Processor (DPP) architecture which enables RFID Anywhere to support and process tag data in any format.

Contents

<b>Topic:</b>	<b>page</b>
<a href="#">DPP overview</a>	96
<a href="#">DPP flexibility</a>	97
<a href="#">Using DPPs</a>	98
<a href="#">Summary</a>	99

## DPP overview

A Data Protocol Processor, or DPP, is used to encode and decode raw RFID tag data. DPPs can be created by developers to turn raw RFID tag data, including custom data, into valuable information with business context, making it easier for that data to be used by business logic. A DPP has a number of components.

### Schema

The schema of a DPP, defined as a .NET DataSet class, defines the various data fields, or columns, that represent the business context of the data. For example, a DPP for EPC-encoded tags would have columns in the DataSet for Company Prefix, Asset Reference, and so on, depending on the specific encoding, whereas the schema for a DPP used for ISO 18000-6 tags used for a closed-loop IT asset tracking system could have columns for department name, employee number and so on.

### Decode method

The Decode method of a DPP takes as input the raw RFID tag data directly from the RFID reader, casts it as the appropriate tag type (for example, ISO 18000-6B), and returns a DataSet in the format defined by the DPP's schema. The developer of the DPP defines the process for decoding the information. For example, if a tag has an employee number as part of its custom data, then the Decode method will take the appropriate bytes and place that information in the Employee Number column of the DPP schema.

The Decode method can be used programmatically from a business module, or automatically from within the Report Engine MP to allow developers to work with processed information instead of raw RFID tag data.

### Encode method

The Encode method of a DPP takes a DataSet and converts the values into raw tag information using whatever process and format the developer requires. The result of the Encode method is an object containing raw, encoded tag data that can be written to an RFID tag.

## DPP flexibility

With RFID Anywhere’s DPP capabilities, organizations have the utmost flexibility when choosing which tags to use, and how to encode data on them.

☞ For more information on developing and using DPPs, see “[Creating and Using a Data Protocol Processor \(DPP\)](#)” [*RFID Anywhere Developer’s Guide*, page 65].

### Flexible tag and protocol selection

When performing tag and hardware selection, an organization can choose the tag that is best suited for the application or environment, without having to worry about the data that will ultimately be encoded on the tags. This is possible because a custom DPP can be created to return the appropriate tag data, regardless of how it was originally encoded on the tag.

For example, a company that wants to track items with EPC encodings can choose to write the EPC data to an ISO 18000-6B tag instead of being forced to use EPC tags. The DPP extracts the underlying data format and protocol from the business logic, since by the time the business logic receives the tag data from RFID Anywhere, the DPP has already decoded the raw data and enabled the various fields, such as Company Prefix or even Department Name or Expiration Date, to be accessed easily.

### Flexible data format, including encryption

When choosing how to encode custom data to a tag, organizations have complete flexibility. They can choose any algorithm they want to use to encode their custom data, and can add additional layers of complexity for security. For example, an organization that wants to encode an expiration date in a format that would be unreadable to the naked eye could choose to encode the year in reverse order in bytes 1, 3, 5, and 7, the month in bytes 15 and 16, and the day in bytes 25 and 26. Then, the DPP’s Decode method would be the only one able to retrieve this information. If an organization wanted to encrypt their custom data using an encryption algorithm, this could be done within the Encode and Decode methods as well.

## Using DPPs

Once created and installed into RFID Anywhere, DPPs can be used by business logic running in RFID Anywhere.

### Report Engine MP

The DPP property of a multiprotocol report definition instructs RFID Anywhere which DPP to use for each tag type it is reporting on. For example, a DPP value of ISO180006B:BaggageTrackingDPP instructs the Report Engine MP to use the BaggageTrackingDPP to decode any ISO 18000-6B tag events it receives.

The format of the Queries property of a multiprotocol report definition must match the schema of the DPP. For example, the query `select * from ITAssets where CO = Y` relies on the DPP schema having defined a table name of 'ITAssets' and a column called 'CO' indicating whether or not the asset is owned by the company. This query would return all fields for assets that were company owned, and would not report on employees personal assets.

### Business modules

Custom business modules leverage DPPs to drive any necessary encoding or decoding from within their code. For example, a business module designed to map raw tag data to asset information from a database receives the raw (HEX) tag data from the RFID reader controller and uses the DPP's Decode method to easily access the various fields encoded within the raw data. By doing this, the business logic in the business module can refer to specific fields of importance, such as the tag's Company Prefix or Asset Reference, without getting bogged down by the details of the HEX representation or other format.

## Summary

This chapter introduced the concept of a Data Protocol Processor, or DPP, which can be created by developers to enable the encoding and decoding of RFID tag data. The plug-and-play design of the DPP framework enables custom DPPs to be easily installed in, and used by, the various RFID Anywhere development models. By using DPPs, users have the utmost flexibility when it comes to tag selection and design, and developers are provided with a unique toolset to simplify the processing of RFID tag data.



---

## CHAPTER 7

# RFID Anywhere Messaging Connectors

About this chapter

This chapter provides detailed information on all of the messaging connectors that come with RFID Anywhere.

Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Introduction</a>	102
<a href="#">Adding and configuring connectors</a>	104
<a href="#">Installing future connectors</a>	105
<a href="#">File messaging connector</a>	106
<a href="#">MSMQ messaging connector</a>	110
<a href="#">QAnywhere messaging connector</a>	113
<a href="#">SMTP messaging connector</a>	118
<a href="#">TCP messaging connector</a>	121
<a href="#">UDP messaging connector</a>	124
<a href="#">Web Service messaging connector</a>	127
<a href="#">Troubleshooting</a>	130

## Introduction

RFID Anywhere features a componentized architecture that is future-proof, plug-and-play oriented, and an easy to use development platform. An essential component of RFID Anywhere is the connectors. Connectors implement the protocol and abstract native features of hardware devices and communication protocols. There exists a connector for a broad range of devices and communication protocols. The concept of connectors proves advantageous in many aspects of an RFID implementation. They insulate developers from low-level hardware interfaces, they can be installed and uninstalled with ease, they are customized for each hardware device so features that are unique to a particular hardware are not discarded, and they can be dynamically configured via the Administrator Console's graphical user interface.

RFID Anywhere has two types of connectors: hardware and messaging. Hardware connectors interact with a specific hardware device, such as an RFID EPC reader, and allow administrators to configure it from the Administrator Console. Messaging connectors abstract low-level aspects of a communication protocol such as TCP or SMTP, and enable RFID Anywhere to easily send processed information to enterprise systems and applications. The focus of this chapter is on messaging connectors.

There are seven different messaging connectors that are available with RFID Anywhere. They include:

- ◆ File messaging
- ◆ MSMQ
- ◆ QAnywhere
- ◆ SMTP
- ◆ TCP/IP
- ◆ UDP
- ◆ Web service

These connectors can be added and configured via a graphical user interface from the Administrator Console. When iAnywhere releases future messaging connectors, the plug-and-play model of RFID Anywhere allows administrators to add them without disturbing their existing RFID infrastructure.

Developers do not need to hard code any reference to messaging connectors when compiling a business module. The business module needs only to be

concerned with generating the output data in the correct format. The list of messaging connectors to send the processed information through can be specified via the Administrator Console. This makes business modules future-proof and enables them to easily communicate with a wide variety of enterprise applications.

RFID Anywhere messaging connectors (with the exception of web service messaging) have the ability to stream data to support sending large volumes of data.

This chapter discusses the seven messaging connectors in detail. For each connector, the purpose, advantages, and the prerequisites are described. This is followed by a detailed explanation of all the properties found in the Administrator Console. A troubleshooting section at the end of this chapter provides tips on how to resolve problems related to messaging connectors.

Before getting into specific messaging connectors, however, general topics such as adding and configuring existing connectors and installing future connectors are discussed.

## Adding and configuring connectors

All connectors that are installed, whether they be messaging or hardware connectors, can be viewed from the Service Type list in the Administrator Console. Connectors need to be added to the Local Services list before they can be configured or started.

For the purpose of this tutorial, assume that you want to add and configure a TCP messaging connector for sending processed information to a departmental application listening for messages on a specific network port. The steps described below apply for all messaging connectors, not just for the TCP connector.

### ❖ To add connectors to Local Services list

1. In the Administrator Console Service Type dropdown list, choose TCP Messaging, and then click Add.

A TCP Messaging connector is added to the Local Services list. The default name of the connector is TcpIpConnector1.

### ❖ To configure connector properties

1. Select the TCP Messaging connector in the Local Services list, and then click Edit.

The Properties Manager for the TCP Messaging connector appears to the right.

From the Properties Manager, you can edit all of the available properties of this messaging connector. A detailed description of the properties for each messaging connector is provided in the sections below.

### ❖ To start connectors

1. Select the configured TCP messaging connector from the Local Services list, and then click Start.

The TCP messaging connector is started.

Once the TCP messaging connector is started, processed information is sent to a business module or an external application listening on the specified port.

## Installing future connectors

Messaging connectors leverage RFID Anywhere's plug-and-play architecture for new component discovery. As iAnywhere develops future messaging connectors, installing them to your RFID network is quick and easy, and it can be done without disturbing the rest of your RFID infrastructure.

New RFID Anywhere connectors will be available as MSI files. As described below, when a new messaging connector is developed, installing the connector to your existing RFID Anywhere solution is as easy as deploying the MSI package using the Provisioning tab of the Administrator Console.

☞ For instructions on deploying packages to RFID Anywhere through Provisioning, see [“Deploying packages” on page 47](#).

## File messaging connector

File messaging is a method of communication where two or more applications share files that contain valuable data in a common format. Files are stored and retrieved from a storage location that is accessible by all applications.

The File messaging connector is used to send processed information in the form of files to a user-specified location on the local hard drive or on a network drive. It is often used as a tool for troubleshooting, as it simply sends data to a location on the file system and requires little configuration to set up, and output can be viewed with a simple text editor. When used as a troubleshooting tool, the connector can be specified to send reports in text file (.txt) format. Developers, system integrators, or administrators can then simply open the text files to view the contents and debug as needed.

☞ For information about using the File Messaging connector as a troubleshooting tool, see [“Troubleshooting” on page 130](#).

When using the File messaging connector, users need to be aware of the space available on the local hard drive or network drive. Since reports may be configured to be generated as frequently as every second, files could be created at a very rapid pace. If the connector runs long enough or if the output is large, insufficient disk space could cause the system to crash.

### Requirements

Depending on the number of reports expected to be generated and their respective sizes, there must be sufficient space on the local hard drive or network drive.

The properties of the File messaging connector can be configured in the Administrator Console.

### Properties

The following table lists and describes all of the properties for the File messaging connector found in the Administrator Console. It also contains examples where applicable.

Property	Description
Append	Determines whether or not the file is appended if it already exists. For example, if the date format creates a file every day it would make sense to append to that file. The default value is True.
DateFormat	<p>Specifies the optional date format added to the file name format. The standard .NET date time format strings are used:</p> <ul style="list-style-type: none"> <li>◆ Day = <i>dd</i></li> <li>◆ Month = <i>mm</i></li> <li>◆ Year = <i>yyyy</i></li> <li>◆ Hour = <i>hh</i></li> <li>◆ Minute = <i>mm</i></li> <li>◆ Seconds = <i>ss</i></li> <li>◆ Fraction of a second = <i>ffffff</i></li> <li>◆ Universal Date Time = <i>Z</i></li> </ul> <p>The default value is <i>dd-MM-yyyy-hh-mm-ss-ffffff</i>.</p>
Description	A description of the service as you want it to appear in the Local Services list. The default description is File Messaging.
Directory	<p>The Destination directory for notification files. If the location does not exist, the connector attempts to create it.</p> <p>Allowable values are local directories, and network paths such as <code>\\server\mydrive</code>. The default location is <code>c:\My Files</code>.</p>

Property	Description
FileName	<p>Specifies the name of the file that will be created or appended to when a notification is received. The file name can contain an additional “{0}” to specify that date and time information should be included. Date and time information is optional, so a fixed file name can also be used.</p> <p>The file extension can be changed to any value that external application(s) understand.</p> <p>If the connector is used as a troubleshooting tool, .txt is the recommended file extension. It will allow you to open the files and view the contents using a standard text editor. The default value is <i>file-{0}.txt</i>.</p>
Name	<p>The service’s name. It is used to refer to this service from other services and it is the name that appears in the Local Services list. The default value is FileConnector1 or FileConnector2, and so on, depending on the number of File messaging connectors already added to the Local Services list.</p>
Version	<p>The version of the service.</p>
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>

Property	Description
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Status	The status of the service.
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## MSMQ messaging connector

MSMQ is a queue-based communication protocol and one of its biggest advantages is that it is asynchronous. Much like email, the recipient application does not need to be online or running when data is sent to it. MSMQ stores the message in queues on different, specified computers across the network. When the recipient computer is ready to accept the message, it can tap into the queue and retrieve the data. However, unlike an email, MSMQ does not require an online connection to transmit data. The messages are queued up on the sender's system and the recipient's computer can retrieve them when needed.

There are no restrictions on the format of data sent via MSMQ. It can transmit data in any format as long as both the sender and recipient applications mutually understand them. MSMQ is included as part of Windows XP Professional and Windows 2003.

The MSMQ messaging connector is used to transmit RFID Anywhere information to applications via the MSMQ protocol. Like any other connector, it abstracts low-level details of the MSMQ communication protocol. Thus, as long as the enterprise application is capable of receiving messages, the developer does not need to program MSMQ. The connector is easy to set up from the Administrator Console.

### Requirements

- ◆ MSMQ running on the host and target computers.
- ◆ The name of the computer where MSMQ connector is located.
- ◆ A target enterprise application that is capable of receiving MSMQ messages.
- ◆ The queue name that the target application is looking for. This is much like the Port property for the TCP connector.

These properties of the MSMQ connector can be configured in the Administrator Console.

### Properties

The following table lists and describes all of the properties of the MSMQ connector found in the Administrator Console. It also contains examples where applicable.

Property	Description
Computer Name	The name of the computer the MSMQ service is located on. If the service is running locally, leave it as the default value: localhost.
Description	A description of the service that appears in the Local Services list. The default value is MSMQ Messaging.
Name	The name of the service. It is used to refer to this service from other services and is the name that appears in the Local Services list. The default value is MSMQConnector1 or MSMQConnector2, and so on, depending on the number of MSMQ messaging connectors already added to the Local Services list.
Private Queue Name	The private queue name. This the name given to the queues sent by this service. An external application must refer to this name in order to receive the queues. For example, RATestQueue. The default value is RFIDAnywhere.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>

Property	Description
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Status	<p>The status of the service.</p>
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## QAnywhere messaging connector

QAnywhere is a messaging solution that is available with SQL Anywhere Studio from iAnywhere. It is a queue-based store and forward messaging technology. QAnywhere contains transmission rules to optimize performance and costs, has encryption during storage and transmission, and supports message compression. It can be enabled to operate in push mode so that mobile users are able to receive valuable RFID data in real-time.

The QAnywhere messaging connector allows RFID Anywhere components to transmit data to mobile applications via QAnywhere. It requires QAnywhere to be running on the host computer and mobile device(s). Using the QAnywhere messaging connector, rich RFID data can be securely and reliably sent to multiple mobile applications running on several mobile devices.

### Note

The QAnywhere messaging connector requires SQL Anywhere to be installed, and must be downloaded and installed separately.

## Requirements

- ◆ QAnywhere running on the host computer and mobile device(s).
- ◆ The queue name that the target application is looking for. This is much like the Port property of TCP connector.
- ◆ Message storage ID.

These properties of the QAnywhere Messaging connector can be configured in the Administrator Console.

## Properties

The following table lists and describes all of the properties of the QAnywhere Messaging connector found in the Administrator Console. It also contains examples where applicable.

Property	Description
Compression Level	Sets the compression level, where the possible values are None, Medium and High. The default value is None.

Property	Description
Connection Parameters	<p>A connection string used by the QAnywhere Manager to connect to the message store database. The connect options are written in the form <i>keyword=value</i>.</p>
Connector Retry Timeout	<p>The time at which send retries through the connector are stopped and the send is failed. The default value is 0 to indicate that the connector will not attempt to send a retry message.</p> <p>This property maps to QAnywhere's MessageProperties::RETRY_TIMEOUT variable. See the QAnywhere documentation for more information.</p>
Description	<p>The description of the connector that appears in the Local Services list. The default value is QAnywhere Messaging.</p>
Connector Max Retry	<p>An optional property for messages sent through a connector. The maximum number of send retries at the connector before failing the send. The default value is 0 to indicate that the connector should not retry sending the message.</p> <p>This property maps to QAnywhere's MessageProperties::RETRY_MAX variable. See the QAnywhere documentation for more information.</p>
Message Storage ID	<p>The message store ID. By default, this is the name of the computer hosting the QAnywhere messaging connector.</p>

Property	Description
Name	The name of the connector that appears in the Local Services list. This is the name you use to refer to this connector from other local services. The default value is QAnywhereConnector1 or QAnywhereConnector2, and so on, depending on the number of QAnywhere messaging connectors already added to the Local Services list.
Message Priority	The priority of the message. The value can be one of Low, Normal, or High. The default value is Normal.
Queue Name	The Queue name. The mobile application must be listening for the same Queue name. The default value is RFIDAnywhere.
Connector Retry Failed Address	An optional property for messages sent through a connector. Once either the RetryMax or RetryTimeout is exceeded, if this property is set, the message is sent to this address.
Connector Retry Failed Priority	An optional property for messages sent through a connector. If a message is sent to the RetryFailedAddress, the message priority is set to this. The value can be one of Low, Normal, or High. The default value is Normal.
Version	The version of the service.

Property	Description
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Status	<p>The status of the service.</p>

Property	Description
Trace Level	<p data-bbox="834 248 1170 361">This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul data-bbox="834 383 1201 748" style="list-style-type: none"><li data-bbox="834 383 1201 439">◆ <b>Off</b> No messages are written to the log file.</li><li data-bbox="834 453 1201 534">◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li><li data-bbox="834 548 1201 609">◆ <b>Warning</b> Only warning messages are written to the log file.</li><li data-bbox="834 623 1201 678">◆ <b>Info</b> Only informational messages are written to the log file.</li><li data-bbox="834 692 1201 748">◆ <b>Verbose</b> All messages are written to the log file.</li></ul>

## SMTP messaging connector

The Simple Mail Transfer Protocol or SMTP is the protocol used to send mail messages between servers. It is generally used to send emails from a mail client to a mail server.

The SMTP messaging connector allows you to send RFID Anywhere information to email accounts via an SMTP mail server. You can specify the recipients using the Administrator Console.

The SMTP messaging connector is especially useful if you need to send alerts or reports to people that are on the move. Whether they are using mobile devices such as a RIM Blackberry, or laptops with an Internet connection, they can receive important RFID information away from their offices. An advantage of sending reports via email is that the recipient does not always need to be connected to the Internet. Whenever an Internet connection is available, the recipient can simply log into their email account and download the RFID Anywhere content.

### Requirements

- ◆ An active SMTP server to send emails.
- ◆ The IP address or the host name of the SMTP server and the port that it is listening on.
- ◆ Recipient email address.
- ◆ Sender email address.

These properties of the SMTP connector can be configured in the Administrator Console.

### Properties

The following table lists and describes all of the properties of the SMTP connector found in the Administrator Console. It also contains examples where applicable.

Property	Description
Description	A description of the service that appears in the Local Services list. The default value is SmtP Messaging.

Property	Description
Name	<p>The name of the service. The name is used to refer to this service from other services and it is the name that appears in the Local Services list. The default value is SmtConnector1 or SmtConnector2, and so on, depending on the number of SMTP messaging connectors already added to the Local Services list.</p>
Port	<p>The port that the SMTP server is listening on. The default port is 25.</p>
Recipients	<p>Email addresses of the SMTP message recipients. Multiple recipients should be delimited by semicolons. For example:</p> <pre data-bbox="834 756 1170 803">recipient1@yourserver.com; recipient2@yourserver.com</pre>
Sender	<p>A valid SMTP sender email address. This appears as the From address in generated email messages. For example, sender@yourserver.com.</p>
Server	<p>The SMTP server's IP address or host name. For example, smtp.yourserver.com.</p>
Version	<p>The version of the service.</p>
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li data-bbox="834 1225 1198 1364">◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li data-bbox="834 1381 1198 1520">◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>

Property	Description
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Status	The status of the service.
Subject	<p>The SMTP message subject header. This appears as the subject of the email. For example:</p> <p>This is a test message sent from RFID Anywhere</p>
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## TCP messaging connector

The TCP communication protocol is a very reliable and safe mode of data transmission. It is a connection-oriented transport, which means that the sender (the RFID Anywhere TCP connector) and the receiver (an external application) acknowledge each other before transporting data. It allows for two-way data transfer (full duplex), and like a file, it transmits data as a stream of bytes and hides the datagram nature of the network. The TCP protocol can detect and resolve network congestion and it can arrange for retransmission if data gets lost. It is not as fast as the UDP protocol, but the additional features often justify using the TCP protocol in certain situations.

The TCP messaging connector is used to send RFID Anywhere content to an external application via the TCP protocol. Like any other messaging connector offered by RFID Anywhere, the TCP messaging connector hides low-level properties and can be set up very easily from the Administrator Console.

### Requirements

- ◆ The IP address of the computer hosting the external application or business module.
- ◆ The port that the application is listening on.

These properties of the TCP connector can be configured in the Administrator Console.

### Properties

The following table lists and describes all of the properties of the TCP connector found in the Administrator Console. It also contains examples where applicable.

Property	Description
IP Address	The Ethernet IP address or host name. This is the address of the computer that is hosting the external application. If the application is running locally, enter localhost.
Description	A description of the service as you would like to appear under the Local Services list. The default value is Tcp Messaging.

Property	Description
Name	The name of the service. It is used to refer to this service from other services and it is the name that appears in the Local Services list. The default is TcpIpConnector1 or TcpIpConnector2, and so on, depending on the number of TCP messaging connectors already added to the Local Services list.
Port	The Ethernet port. This is the port that the connector will send output and reports to. An external application should listen on this port to receive the data. The default value is 0.
Retries	The number of tries to send a message to an end point. If all retries fail, the accumulated messages will be removed. The default value is 5.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>

Property	Description
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Status	The status of the service.
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## UDP messaging connector

The User Datagram Protocol (UDP) is faster than TCP and it is most often used in applications that demand real-time responses. It ensures data integrity by performing checksums. It is a connectionless protocol, which means that the sender (RFID Anywhere UDP connector) and the receiver (an external application) do not acknowledge each other before transmitting the data, and delivery is not guaranteed. UDP data is smaller in size than that of TCP, and unlike TCP, they are not responsible for resolving errors that may occur during transmission. This reduced size and freedom allow UDP to communicate at a much faster rate.

Because of the lack of error checking, there is no way to detect network congestion and a datagram could arrive out of order or even get lost, so the end application must be capable of detecting these issues and splitting the data into appropriate packets. Nonetheless, the UDP protocol is used in real-time applications where the speed of data transmission is a concern.

The UDP messaging connector is used to send RFID Anywhere content to an external application using the User Datagram Protocol (UDP).

### Requirements

- ◆ An external application that is capable of accepting UDP messages and resolving any errors that may occur during transmission.
- ◆ The Ethernet IP address or host name of the computer hosting the external application.
- ◆ The port that the external application is listening on.

These properties of the UDP connector can be configured in the Administrator Console.

### Properties

The following table lists and describes all of the properties of the UDP connector found in the Administrator Console. It also contains sample entries where applicable.

Property	Description
IP Address	The Ethernet IP address or host name. This is the address of the computer that is hosting the external application. If the application is running locally, enter localhost.

Property	Description
Description	A description of the service that appears in the Local Services list. The default value is Udp Messaging.
Name	The name of the service. It is used to refer to this service from other services and it is the name that appears in the Local Services list. The default value is UdpConnector1 or UdpConnector2, and so on, depending on the number of UDP messaging connectors already added to the Local Services list.
Port	The Ethernet port. This is the port that the connector sends output and reports to. An external application should listen on this port in order to receive the data. The default value is 0.
Retries	The number of tries to send a message to an end point. If all retries fail, the accumulated messages are removed. The default value is 5.
Version	The version of the service.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>

Property	Description
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Status	<p>The status of the service.</p>
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"> <li>◆ <b>Off</b> No messages are written to the log file.</li> <li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li> <li>◆ <b>Warning</b> Only warning messages are written to the log file.</li> <li>◆ <b>Info</b> Only informational messages are written to the log file.</li> <li>◆ <b>Verbose</b> All messages are written to the log file.</li> </ul>

## Web Service messaging connector

Web services is a software system designed for interoperability between applications running on different computers. Each web service has a unique Web Service Definition Language (WSDL), which is essentially an XML file that contains web service-related information, such as its location and a list of supported methods. To access a method supported by a web service, an application must generate messages as specified by the WSDL. The messages may need to be enclosed in a SOAP envelope before they are transmitted to the web service using HTTP.

The advantages of web services are that they use open standards and protocols, they support all operating systems, and they can work securely through corporate firewalls.

### Requirements

- ◆ A live web service that can be securely accessed.
- ◆ A web link to the Web Service Definition Language (WSDL) file for the web service.
- ◆ A web method to call.

These properties of the Web Service connector can be configured in the Administrator Console.

### Properties

The following table lists and describes all of the properties of the Web Service connector found in the Administrator Console. It also contains examples where applicable.

Property	Description
Description	A description of the service that appears in the Local Services list. The default value is WebService Messaging.
Web Method	The web service method to be called. A list of all of the web methods supported by the web service can be obtained from the WSDL file.

Property	Description
Name	<p>The name of the service. It is used to refer to this service from other services and it is the name that appears in the Local Services list. The default value is WebServiceConnector1 or WebServiceConnector2, and so on, depending on the number of Web Service messaging connectors already added to the Local Services list.</p>
Version	<p>The version of the service.</p>
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>
Start Operation	<p>If the Start Mode property for the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>

Property	Description
Status	The status of the service.
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"><li>◆ <b>Off</b> No messages are written to the log file.</li><li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li><li>◆ <b>Warning</b> Only warning messages are written to the log file.</li><li>◆ <b>Info</b> Only informational messages are written to the log file.</li><li>◆ <b>Verbose</b> All messages are written to the log file.</li></ul>
WSDL	A web link to a Web Services Description Language (WSDL) file for the web service. For example, <a href="http://test.testwsdl.com/WSDL">http://test.testwsdl.com/WSDL</a> .

# Troubleshooting

This section contains general troubleshooting steps that apply to all messaging connectors.

## Using a File messaging connector to troubleshoot

RFID Anywhere solutions often involve multiple components that include several connectors, business modules, and external applications. When RFID Anywhere processed information (reports) are not being received by an external application, the first step in troubleshooting is to determine the component that is malfunctioning. The following steps illustrate how you can use the File messaging connector to narrow down the source of the problem.

### ❖ To determine if a messaging connector is the problem

1. In the Administrator Console Home page, choose File Messaging from the Services Type dropdown list. Click Add.  
A File messaging connector is added to the Local Services list.
2. Select the newly-added File messaging connector, and then click Edit.  
The Properties Manager pane for the File messaging connector opens.
3. Click the pencil beside the Directory property.
4. Enter the directory to where you would like to send the files, or use the default value `c:\My Files`.
5. Ensure that for the File Name property of the File Messaging connector, the file extension is `.txt`.
6. Add the File messaging connector as a subscriber to all of the necessary components, such as custom business modules, ALE Event Cycle connectors, and MultiProtocol Report Definition connectors.
7. Start all of the required services, including the File messaging connector.
8. Open the directory specified in the Directory property of the File messaging connector. Check to see if any files are being created.  
If files are not being created, then you may need to change some code or settings in your services other than the messaging connector. Since the File messaging connector is the simplest messaging connector to set up and use, this would be a strong indication that the messaging connector is not the problem.


If files are being generated, then open one or more of the files to view the content. You can open the files using a simple text editor.

If the file contents appear to be correct, then you can be certain that your custom business modules, ALE Event Cycle connectors, and/or MultiProtocol Report Definition connectors are configured properly. In this case, the problem is most often either with the messaging connector or the external application that is receiving the data.

## Using Monitoring to troubleshoot

The Monitoring page in the Administrator Console allows you to see errors reported by messaging connectors. The connectors can be configured to report all messages, or filtered to report only error messages, warning messages, or informational messages.

Monitoring can be used to report errors from all types of connectors and business modules.

 For more information about Monitoring, see [“Monitoring an RFID Anywhere network” on page 43](#).

### ❖ To use Monitoring to troubleshoot

1. In the Administrator Console Home page, select the messaging connector you want to troubleshoot. Click Edit.

The Properties Manager for the connector opens.

2. Click the pencil beside the Trace Level property.
3. Choose Verbose from the dropdown list to report all messages, and the click the green checkmark to save the data.

The following table contains descriptions of all the options available in the dropdown list.

Trace Level

This property specifies the level of information written to the log file. The value can be one of the following:

- ◆ **Off** No messages are written to the log file.
- ◆ **Error** Only error messages are written to the log file. This is the default value.
- ◆ **Warning** Only warning messages are written to the log file.
- ◆ **Info** Only informational messages are written to the log file.
- ◆ **Verbose** All messages are written to the log file.

4. Click Monitoring.

The Monitoring page opens.

5. From the System Monitoring dropdown list, choose the messaging connector you want to troubleshoot.

All of the events pertaining to the connector appear.

6. From the Events dropdown list, choose the date.

The list of events is filtered to display only the events that occurred on the specified date.

From the list of events, you can read the messages sent from the connector and troubleshoot the problem accordingly.

PART II

# RFID ANYWHERE TUTORIALS

This part contains tutorials to help you get started using all the components of RFID Anywhere.



---

## CHAPTER 8

# About the Tutorials

RFID Anywhere provides a variety of options for applying business logic, filtering, and grouping to raw RFID data: Application Level Events (ALE), custom business module development, and Report Engine MP. The tutorials will introduce you to these methods and help you get started in applying your own business logic.

### Tutorial format

- ◆ **Creating Simulated RFID Data** Introduces RFID Simulator Data Editor and illustrates the steps involved in creating a complete tag simulation based on a real-life scenario. You will then create and configure a Simulator connector to use with the rest of the tutorials.
- ◆ **Using ALE to Filter and View RFID Data** Introduces the Application Level Events standard and shows how to generate processed information (XML reports) that adhere to the ALE specification.
- ◆ **Using Report Engine MP** Introduces the Report Engine MP and shows how to generate processed information in the form of XML reports.
- ◆ **Inventory Tracker Demo** Introduces the Inventory Tracker demo package that is included with RFID Anywhere. The chapter provides an introduction to custom business modules, and shows examples of applying all three development models in real-world scenarios.

☞ To learn more about custom business module development and how to create your own business modules, see “[Using the RFID Anywhere Visual Studio Extension to Create Custom Business Modules](#)” [*RFID Anywhere Developer’s Guide*, page 7].



---

## CHAPTER 9

# Creating Simulated RFID Data

### About this chapter

This chapter shows you how to use the RFID Anywhere Simulator Data Editor to simulate an RFID network and create simulation files. It then takes you through the steps of configuring an RFID Multiprotocol Simulator connector that makes use of the simulation files.

### Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Introduction</a>	138
<a href="#">Lesson 1: Designing the Tag Activity timeline</a>	139
<a href="#">Lesson 2: Using the RFID Simulator Data Editor to create simulation data</a>	142
<a href="#">Lesson 3: Configuring the RFID Multiprotocol Simulator connector</a>	150
<a href="#">Summary</a>	152

## Introduction

In this tutorial, you use the RFID Anywhere Simulator Data Editor to simulate an RFID network. This tutorial demonstrates how you can simulate an RFID network to build and test applications without actual RFID hardware.

The RFID Anywhere Inventory Tracker package ships with sample simulation data. This chapter outlines the steps for designing and creating simulation data for use with the Inventory Tracker demos. The steps and the lessons learned in this chapter can be used to create simulation data for any scenario.

This tutorial also shows you how to create and configure an RFID Multiprotocol Simulator connector that you can use in other tutorials.

Tasks you will complete in this tutorial include:

- ◆ Designing the tag activity timeline
- ◆ Using the RFID Simulator Data Editor to create simulation data
- ◆ Configuring the RFID Multiprotocol Simulator connector

## Lesson 1: Designing the Tag Activity timeline

Before using the RFID Simulator Data Editor to build the actual simulation files, it is useful to design a timeline of tag activity for the scenario in a spreadsheet or other tool. This will allow you to visualize tag groupings easily. Tags are grouped according to time with RFID Anywhere's simulation environment, so tags that appear at the same time and remain for the same duration would be part of one group. The following screenshot shows the tag activity timeline for the Inventory Tracker sample.

	A	B	C	D	E	F	G	H	I	J	K
		<b>Tag Data</b>			<b>Serial Numbe</b>		<b>Time 0</b>	<b>Time+10</b>	<b>Time+20</b>	<b>Time+30</b>	<b>Time+40</b>
1		<b>Source1</b>									
2		1.1.2.5	Robertson Health	AutoClean 77 Sterilizer	5						
3		1.1.2.6	Robertson Health	AutoClean 77 Sterilizer	6						
4		1.1.2.7	Robertson Health	AutoClean 77 Sterilizer	7						
5		1.1.5.3	Robertson Health	M100 Air Lift Stool	3						
6		1.1.5.8	Robertson Health	M100 Air Lift Stool	8						
7		1.1.8.1	Robertson Health	Cardiograph XL500	1						
8		1.1.25.5	Robertson Health	Cardiology Stethoscope C4	5						
9		1.1.30.2	Robertson Health	499i Centrifuge	2						
10		1.3.4.8	ProHealthOne	JL-23 Adult Portable Scale	8						
11		1.3.6.9	ProHealthOne	LH-26 Physician Digital Scale	9						
12		1.8.34.2	MobileDiag	Hand-held Capnograph/Oximeter	2						
13		1.9.22.3	MobilityFirst	Pedlar Light Workout Exerciser	3						
14		<b>Source2</b>									
15		1.2.7.5	GeneKare	GK600e Folding Emergenos Transpo	5						
16		1.2.7.6	GeneKare	GK600e Folding Emergenos Transpo	6						
17		1.2.7.7	GeneKare	GK600e Folding Emergenos Transpo	7						
18		1.2.20.3	GeneKare	GK400h Heavy Duty Transport	3						
19		1.2.20.4	GeneKare	GK400h Heavy Duty Transport	4						
20		1.2.20.5	GeneKare	GK400h Heavy Duty Transport	5						
21		1.2.20.6	GeneKare	GK400h Heavy Duty Transport	6						
22		1.2.20.7	GeneKare	GK400h Heavy Duty Transport	7						
23		1.12.65.10	InnovaRide	Rx300 Manual Wheelchair	10						
24		1.12.65.11	InnovaRide	Rx300 Manual Wheelchair	11						
25		1.12.65.12	InnovaRide	Rx300 Manual Wheelchair	12						
26		1.30.21.1	Flow Guard Inc.	6060Iv IV Pump	1						
27		1.30.21.2	Flow Guard Inc.	6060Iv IV Pump	2						
28		1.7.21.5	Assist Team	Oxygen Cart Aluminum	5						
29		1.5.8.2	Baltic Labs	Alternating Pressure Air Mattress	2						
30		1.9.22.3	MobilityFirst	Pedlar Light Workout Exerciser	3						

Recall that two simulation files are used in that sample, indicated by the sections Source1 (for the first simulation file) and Source2 (for the second simulation file). Each column in the timeline spreadsheet represents data that is important to either the creation of the simulation files, or in mapping the simulated data to meaningful business information as follows:

### Column B: Tag Data

This column represents the unique RFID tag information that is used to represent a unique asset. When designing a simulation, it is important to understand the tag encoding that will be used, and the different components of a tag ID for that encoding.

Here, the EPC encoding SGTIN-96 partition SIX is used. Tag IDs for this encoding consist of four parts: Filter, Company Prefix, Item Reference, and Serial Number. The tag ID in row 16, which is 1.2.7.5, denotes a CompanyPrefix of 2 (GeneKare), an ItemReference (SKU) of 7 (GK600e

Folding Emergency Transport), and a Serial Number of 5 (unique Serial Number of the specific item).

A variety of Company Prefixes, Item References, and Serial Numbers are represented with this tag activity timeline.

### Column C: Company

This column provides a mapping between the CompanyPrefix in column B to the corresponding meaningful (albeit imaginary) company name. Note that all rows where the Tag Data has a CompanyPrefix matching another row will have the same Company value (such as rows 3 through 10). This information is ultimately used by the business module or enterprise application to map the raw tag data to meaningful information.

### Column D: SKU

This column provides a mapping between the ItemReference in column B to the corresponding meaningful (albeit imaginary) product name or SKU. Note that all rows where the Tag Data has an ItemReference matching another row will have the same SKU value (such as rows 19 through 23). This information is ultimately used by the business module or enterprise application to map the raw tag data to meaningful information.

### Column E: Serial Number

This column clearly identifies the unique Serial Number of each asset. It is important to ensure that each Serial Number for a given ItemReference is unique, since serialized data is at the heart of an RFID environment (such as rows 24 through 26).

### Column G: Time 0

In this tag timeline, tag groups have been broken into 10 second groupings. Column G represents Time 0 (the time the simulation is started) until Time +10 seconds (10 seconds after the simulation has started). Here, any row with Column G filled in is be part of the first group of tags created in the RFID Simulator Data Editor for a given source. For Source1, this indicates that tags 1.1.2.5, 1.1.2.6, and 1.3.6.9 appear immediately once the simulation starts.

### Column H: Time +10

As with Column G, any row with Column H filled in is part of the second group of tags for the given source. For Source 1, this indicates that tags 1.1.2.5, 1.1.2.6, 1.1.2.7, 1.1.5.3, 1.1.5.8, and 1.1.8.1 will appear 10 seconds

after the simulation is started. Note that tags 1.1.2.5 and 1.1.2.6 appeared in the first group, meaning that for the purposes of this simulation, these tags never disappeared from the simulated RFID reader's read zone. Also note that tag 1.3.6.9, which was part of the first group, is not part of this second group, meaning that it disappeared from the simulated RFID reader's read zone.

### **Columns I through K: Time +20 through Time +40**

As with Columns G and H, any row with one of these columns filled in is part of an additional group of tags that are 'seen' by the simulated reader during a given time period. This complete tag activity timeline spans 50 seconds, and simulates tags that appear for multiple periods before being removed for the rest of the simulation (such as row 3), tags that do not appear until midway through the simulation (such as row 11), and even tags that appear in both sources (such as rows 14 and 31 represent the same physical tag). Note that with the tag that appears in both sources, the tag never appears in both locations at the same time.

## Lesson 2: Using the RFID Simulator Data Editor to create simulation data

Once the tag activity timeline for the scenario has been designed, this information can easily be entered into the RFID Simulator Data Editor and saved to *.smt* files that can be imported and used by the RFID Simulator connector within RFID Anywhere.

The steps below show you how to create a simple simulation file that can be used in place of the *reader1.smt* simulation data file that is included with the Inventory Tracker sample. The simple tag activity timeline used to create this the simulation data in this tutorial is below, followed by the implementation steps.

	A	B	C	D	E	F	G	H	I	J	K
1		<b>Tag Data</b>	<b>Company</b>	<b>SKU</b>	<b>Serial Number</b>		<b>Time 0</b>	<b>Time+10</b>	<b>Time+20</b>	<b>Time+30</b>	<b>Time+40</b>
2		<b>Source1</b>									
3		1.1.2.5	Robertson Health	AutoClean 77 Sterlizer	5						
4		1.1.2.6	Robertson Health	AutoClean 77 Sterlizer	6						
5		1.1.2.7	Robertson Health	AutoClean 77 Sterlizer	7						
6		1.1.5.3	Robertson Health	M100 Air Lift Stool	3						

### Starting the RFID Simulator Data Editor

❖ **To start the RFID Simulator Data Editor**

1. Choose Start ► Programs ► RFID Anywhere ► RFID Simulator Data Editor.

The RFID Simulator Data Editor appears.

### Creating the group of tags for Time 0

❖ **To create the group of tags for Time 0**

1. From the Session menu, choose New to create a new session.
2. From the Groups menu, choose New ► EPC to create the first group of tags (Time 0 in the tag activity timeline).
3. Enter **T0** in the Name field.
4. Enter **2** in the Quantity field.
5. Select Enter Tags Manually in the Creation Mode section.

This allows you to define non-sequential tag data using a simple template.

6. Ensure that Increment Tags Per Session is cleared.

If this option is selected, the Serial Number of each tag is incremented every time the simulation cycle runs. This functionality is not applicable for the Inventory Tracker scenario, but is useful for simulating a conveyor belt or production line scenario.

7. Choose EPC\_SGTIN96 from the Type dropdown list in the Encoding section and choose SIX from the Partition dropdown list.
8. In the Value column of the Filter row in the template, enter **1**.
9. In the Value column of the CompanyPrefix row, enter **1**.
10. In the Value column of the ItemReference row, enter **2**.
11. In the Value column of the SerialNumber row, enter **5**.
12. In the Increment column of the SerialNumber row, select the checkbox.  
This indicates that 5 is the starting Serial Number, and other tags from this template have unique Serial Numbers. Here, since there are only 2 tags being created, they will have Serial Numbers 5 and 6.
13. Choose RFID Anywhere DPP from the DPP dropdown list to use RFID Anywhere's default Data Protocol Processor for EPC tags.
14. Click Create Tags to generate these tags and add them to the group.  
Once created, these tags appear in the Tags window on the right side of the RFID Simulator Data Editor.
15. To add this group of tags to the simulator session, choose Add from the Groups menu.  
The Session Items dialog appears.
16. In the Start Interval field enter **0:0** to indicate that this group of tags appears immediately when the simulation starts.  
If you wanted the simulation to wait 10 seconds before this group of tags appeared, you would have to enter 0:10 in the Start Interval field.
17. In the Life Time field, enter **0:10** to indicate that this group of tags will remain in the simulated reader's read zone for 10 seconds. After 10 seconds, the next group of tags in the sequence, configured in the following steps, will appear.
18. Choose RFID Anywhere DPP from the DPP dropdown list.
19. Choose iAnywhere.RfidNet.Rfid.Tags.RFIDTagEPC\_Class0 from the Class to Generate dropdown list to indicate that RFID Anywhere should generate EPC Class 0 tags from the simulated data.  
These objects are sent to the business logic running in RFID Anywhere.

20. Click OK to save this group of tags to the current session.
21. The group T0 is added to the simulation timeline in the left pane of the RFID Simulator Data Editor.
22. Save the session by choosing Save from the Session menu. Save the file with the name *reader1manual.smt* in the *Inventory Tracker\Simulated Data* folder of your RFID Anywhere installation.

## Creating the group of tags for Time +10

### ❖ To create the tag group for Time + 10

1. From the Groups menu, choose New ► EPC to create the second group of tags (Time +10 in the tag activity timeline).
2. Enter **T1** in the Name field.
3. Enter **3** in the Quantity field.
4. Choose Enter Tags Manually in the Creation Mode section.
5. Ensure that Increment Tags Per Session is cleared.
6. Choose EPC\_SGTIN96 from the Type dropdown list in the Encoding section.  
Choose SIX from the Partition dropdown list.
7. In the Value column of the Filter row in the template, enter **1**.
8. In the Value column of the CompanyPrefix row, enter **1**.
9. In the Value column of the ItemReference row, enter **2**.
10. In the Value column of the SerialNumber row, enter **5**.
11. In the Increment column of the SerialNumber row, select the checkbox.
12. Choose RFID Anywhere DPP from the DPP dropdown list to use RFID Anywhere's default Data Protocol Processor.
13. Click Create Tags to generate these tags and to add them to the group. Once created, these tags appear in the Tags window on the right side of the RFID Simulator Data Editor.
14. To add the second set of tags to this group, you need to change the template after adding the first set of tags since the fourth tag is not in sequence with the first three tags (note that row 6 is for a different ItemReference). Enter **1** in the Quantity field.

15. In the Value column of the Filter row in the template, enter **1**.
16. In the Value column of the CompanyPrefix row, enter **1**.
17. In the Value column of the ItemReference row, enter **5**.
18. In the Value column of the SerialNumber row, enter **3**.
19. Click Create Tags.
20. To add this group of four tags to the simulator session, choose Add from the Groups menu.  
The Session Items dialog appears.
21. In the Start Interval field enter **0:0** to indicate that this group of tags appears immediately after the first group of tags has appeared for their configured Life Time, which is 10 seconds. There is no delay between the two groups of tags.
22. In the Life Time field, enter **0:10** to indicate that this group of tags will remain in the simulated reader's read zone for 10 seconds.
23. Choose RFID Anywhere DPP from the DPP dropdown list.
24. Choose iAnywhere.RfidNet.Rfid.Tags.RFIDTagEPC\_Class0 from the Class to Generate dropdown list to indicate that RFID Anywhere should generate Class0 tags from the simulated data.
25. Click OK to save this group of tags to the current session.
26. The group T1 is added to the simulation timeline in the left pane of the RFID Simulator Data Editor.

## Creating the group of tags for Time +20

### ❖ To create the tag group for Time + 20

1. From the Groups menu, choose New ► EPC to create the second group of tags (Time +20 in the tag activity timeline).
2. Enter **T2** in the Name field.
3. Enter **1** in the Quantity field.
4. Choose Enter Tags Manually in the Creation Mode section.
5. Ensure that Increment Tags Per Session is cleared.

6. Choose EPC\_SGTIN96 from the Type dropdown list in the Encoding section.
7. Choose SIX from the Partition dropdown list.
8. In the Value column of the Filter row in the template, enter **1**.
9. In the Value column of the CompanyPrefix row, enter **1**.
10. In the Value column of the ItemReference row, enter **2**.
11. In the Value column of the SerialNumber row, enter **7**.
12. In the Increment column of the SerialNumber row, select the checkbox.
13. Choose RFID Anywhere DPP from the DPP dropdown list to use RFID Anywhere's default Data Protocol Processor.
14. Click Create Tags to generate this tag and to add it to the group. Once created, this tag appears in the Tags window on the right side of the RFID Simulator Data Editor.
15. To add the second set of tags to this group, you need to change the template. Enter **1** in the Quantity field.
16. In the Value column of the Filter row in the template, enter **1**.
17. In the Value column of the CompanyPrefix row, enter **1**.
18. In the Value column of the ItemReference row, enter **5**.
19. In the Value column of the SerialNumber row, enter **3**.
20. Click Create Tags.
21. To add this group of four tags to the simulator session, choose Add from the Groups menu.  
The Session Items dialog appears.
22. In the Start Interval field enter **0:0** to indicate that this group of tags appears immediately after the previous group of tags has appeared for their configured Life Time, which is 10 seconds. There is no delay between the two groups of tags.
23. In the Life Time field, enter **0:10** to indicate that this group of tags will remain in the simulated reader's read zone for 10 seconds.
24. Choose RFID Anywhere DPP from the DPP dropdown list.

25. Choose `iAnywhere.RfidNet.Rfid.Tags.RFIDTagEPC_Class0` from the Class to Generate dropdown list to indicate that RFID Anywhere should generate Class0 tags from the simulated data. These objects are sent to the business logic running in RFID Anywhere.
26. Click OK to save this group of tags to the current session.
27. The group T2 is added to the simulation timeline in the left pane of the RFID Simulator Data Editor.

## Creating the group of tags for Time +30

1. From the Groups menu, choose New ► EPC to create the second group of tags (Time +30 in the tag activity timeline).
2. Enter **T3** in the Name field.
3. Enter **1** in the Quantity field.
4. Choose Enter Tags Manually in the Creation Mode section.
5. Ensure that Increment Tags Per Session is cleared.
6. Choose EPC\_SGTIN96 from the Type dropdown list in the Encoding section.
7. Choose SIX from the Partition dropdown list.
8. In the Value column of the Filter row in the template, enter **1**.
9. In the Value column of the CompanyPrefix row, enter **1**.
10. In the Value column of the ItemReference row, enter **5**.
11. In the Value column of the SerialNumber row, enter **3**.
12. In the Increment column of the SerialNumber row, select the checkbox.
13. Choose RFID Anywhere DPP from the DPP dropdown list to use RFID Anywhere's default Data Protocol Processor.
14. Click Create Tags to generate this tag and to add it to the group. Once created, this tag appears in the Tags window on the right side of the RFID Simulator Data Editor.
15. To add this group of tags to the simulator session, choose Add from the Groups menu.  
The Session Items dialog appears.

16. In the Start Interval field enter **0:0** to indicate that this group of tags appears immediately after the previous group of tags has appeared for their configured Life Time, which is 10 seconds. There is no delay between the two groups of tags.
17. In the Life Time field, enter **0:10** to indicate that this group of tags will remain in the simulated reader's read zone for 10 seconds.
18. Choose RFID Anywhere DPP from the DPP dropdown list.
19. Choose iAnywhere.RfidNet.Rfid.Tags.RFIDTagEPC\_Class0 from the Class to Generate dropdown list to indicate that RFID Anywhere should generate Class0 tags from the simulated data. These objects are sent to the business logic running in RFID Anywhere.
20. Click OK to save this group of tags to the current session.
21. The group T3 is added to the simulation timeline in the left pane of the RFID Simulator Data Editor.

## Creating the group of tags for Time +40

1. From the Groups menu, choose New ► EPC to create the second group of tags (Time +30 in the tag activity timeline).
2. Enter **T4** in the Name field.
3. Enter **1** in the Quantity field.
4. Choose Enter Tags Manually in the Creation Mode section.
5. Ensure that Increment Tags Per Session is cleared.
6. Choose EPC\_SGTIN96 from the Type dropdown list in the Encoding section.
7. Choose SIX from the Partition dropdown list.
8. In the Value column of the Filter row in the template, enter **1**.
9. In the Value column of the CompanyPrefix row, enter **1**.
10. In the Value column of the ItemReference row, enter **2**.
11. In the Value column of the SerialNumber row, enter **7**.
12. In the Increment column of the SerialNumber row, select the checkbox.
13. Choose RFID Anywhere DPP from the DPP dropdown list to use RFID Anywhere's default Data Protocol Processor.

14. Click Create Tags to generate this tag and to add it to the group. Once created, this tag appears in the Tags window on the right side of the RFID Simulator Data Editor.
15. To add this group of tags to the simulator session, choose Add from the Groups menu.  
The Session Items dialog appears.
16. In the Start Interval field enter **0:0** to indicate that this group of tags appears immediately after the previous group of tags has appeared for their configured Life Time, which is 10 seconds. There is no delay between the two groups of tags.
17. In the Life Time field, enter **0:10** to indicate that this group of tags will remain in the simulated reader's read zone for 10 seconds.
18. Choose RFID Anywhere DPP from the DPP dropdown list.
19. Choose iAnywhere.RfidNet.Rfid.Tags.RFIDTagEPC\_Class0 from the Class to Generate dropdown list to indicate that RFID Anywhere should generate Class0 tags from the simulated data. These objects are sent to the business logic running in RFID Anywhere.
20. Click OK to save this group of tags to the current session.
21. The group T4 is added to the simulation timeline in the left pane of the RFID Simulator Data Editor.

## Save and close the session

1. Save the session by choosing Save from the Session menu. Save the file with the name *reader1manual.smt* to the *Inventory Tracker\Simulated Data* folder of your RFID Anywhere installation.
2. Exit the RFID Simulator Data Editor by clicking Exit from the Session menu.

## Lesson 3: Configuring the RFID Multiprotocol Simulator connector

Now that you have created a simulation file, you need to configure an RFID Multiprotocol Simulator connector to interpret and use the simulation file. The RFID Multiprotocol Simulator connector simulates the appearance of tags that were defined using the RFID Simulator Data Editor. By using a connector, simulated RFID data is fed into RFID Anywhere via the RFID multiprotocol reader controller, in the exact same way that raw data coming from an RFID reader would be. The RFID Multiprotocol Simulator connector uses a simulation file (.smt) to simulate RFID tags being brought into and out of the range of a real reader.

The RFID Multiprotocol Simulator connector configured in this lesson is used in the next several chapters.

**Note**

Although it is recommended that you follow lessons 1 and 2 to create a simulation file, you can follow the instructions below, even if you skipped those lessons.

❖ **To add and configure the RFID MP Simulator connector**

1. From the Start menu, choose Programs ► RFID Anywhere ► Administrator Console.  
The Administrator Console Login page opens.
2. Enter your credentials, and then click Log On.  
The Administrator Console Home page opens.
3. In the Home page, choose **RFID Multiprotocol Simulator** from the Service Type dropdown list, and then click Add.
4. The RFID Multiprotocol Simulator connector is added to the list of local services and is highlighted.  
The default name for the new connector is RfidMPSimulator1. Note that the newly-added connector could be named RfidMPSimulator2 or RfidMPSimulator3, depending on the number of MP Simulator connectors already present under Local Services.
5. To edit the properties of this new connector, select the checkbox beside it and then click Edit.  
The properties for the connector appear in the Properties Manager, to the right of the Local Services list.

6. Expand the Antenna1 set of properties by clicking the + beside Antenna1. Expand the Smoothing set of properties in the same manner.
7. In the textbox for the Glimpsed Timeout property, type **100ms** (milliseconds).
8. Repeat step 2 for the Lost Timeout, Observed Threshold, and Observed Timeout properties.
9. In the textbox for the Cycles per Trigger property, type **1**.
10. In the dropdown list for the Read on Start property, choose **True**.
11. In the textbox for the Name property, type **InvTracSim1**.  
As mentioned earlier, this simulator connector is used again when configuring the Inventory Tracker demo package.
12. Beside the text field for the Session Data property, click Browse and locate the file *reader1manual.smt* located in the *Inventory Tracker\SimulatedData* directory of your RFID Anywhere install, clicking Open when you locate and highlight the file. This is the simulated file that you created in lessons 1 and 2 of this chapter.

**Note**

If you skipped lessons 1 and 2 of this chapter, then select *reader1.smt* located in the *Inventory Tracker\SimulatedData* folder of your RFID Anywhere installation.

13. Click the green disk in the title bar of the Properties Manager to save the changes you made to the RFID Multiprotocol Simulator connector.  
The InvTracSim1 RFID Multiprotocol Simulator connector is updated.
14. Click Refresh to view the updated properties in the Administrator Console.

## Summary

You have now created simulated tags and have configured the RFID Multiprotocol Simulator connector that you can use in the next several chapters. More importantly, you can apply what you have learned in this chapter to create custom tags that simulate any environment.

In the next chapters, you will learn how to generate processed information on RFID EPC tags using a variety of methods.

---

## CHAPTER 10

# Using ALE to Filter and View RFID Data

### About this chapter

This chapter introduces the Application Level Events (ALE) engine and shows how to generate and send processed reports that adhere to the ALE specification.

### Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Introduction</a>	154
<a href="#">Lesson 1: Working with connectors from the Administrator Console</a>	160
<a href="#">Lesson 2: Starting the connectors</a>	168
<a href="#">Lesson 3: Viewing and understanding ALE reports</a>	169
<a href="#">Lesson 4: Stopping the connectors</a>	175
<a href="#">Properties of ALE Event Cycle connectors</a>	176
<a href="#">Summary</a>	181

## Introduction

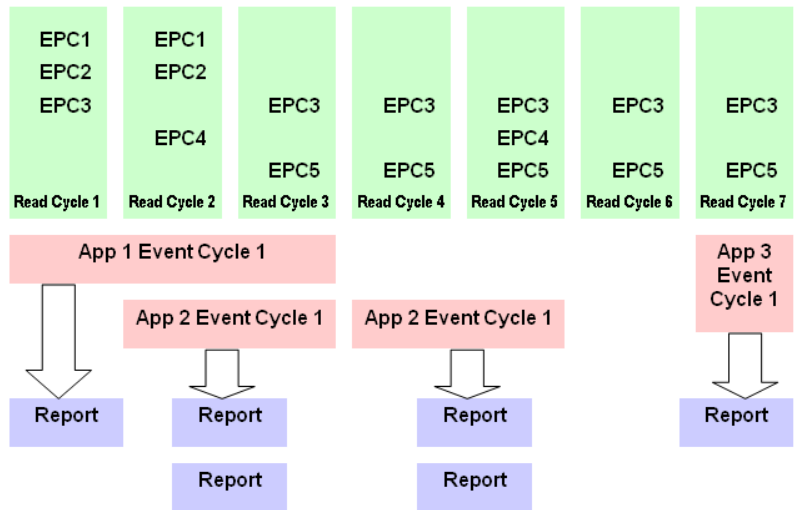
RFID Anywhere includes an **Application Level Events (ALE)** engine provides a way for developers and integrators to process raw RFID data. ALE is a standard defined by EPCGlobal, Inc.

(<http://www.epcglobalinc.org>) that specifies the simple creation of standard XML reports based on the tag activity of EPC tags. These XML reports can then be fed into other enterprise systems by using RFID Anywhere's messaging connectors.

System integrators building RFID systems may find that customers in specific industries require ALE support as part of the solution. Some customers may want a complete ALE solution immediately, while others may want to build a custom solution without sacrificing the ability to deploy ALE-driven applications in the future. System integrators therefore need an extensible platform with support for ALE in addition to other development models. In this light, RFID Anywhere's configurable ALE implementation is both an alternative and a complement to its other development options for building RFID solutions.

### ALE basics

ALE exposes data in the form of periodic, XML-based event cycle reports. An event cycle is simply a series of hardware read cycles: each event cycle report summarizes the tag activity during the read cycles preceding it. The reports are the smallest unit of interaction between the ALE protocol and other applications. The following figure illustrates how a physical reader's read cycles (in green) map to configured ALE event cycles (in red) and the resulting reports (in blue) to be sent to and consumed by an enterprise application. The references to EPC1, and EPC2 to EPC5 represent 5 EPC tags that are read, or seen, by the physical RFID hardware during each read cycle.



When a report is generated and sent to an enterprise application over one of RFID Anywhere's messaging connectors, it is up to the application to add additional business value to the data contained in the report. For example, the application could map the tag data to a specific item or asset for display to a user.

## Understanding the RFID Anywhere ALE implementation

The ALE implementation in RFID Anywhere consists of several key components: the hardware connectors, which communicate with the readers, the **RFID ALE Protocol business module**, which collects raw data from hardware controllers, the **ALE Event Cycle connectors**, which define and validate event cycle reports for the ALE business module, and messaging connectors, which deliver completed reports to enterprise applications for processing.

RFID Anywhere also exposes ALE functionality over web services.

### ALE Event Cycle connectors

RFID Anywhere's ALE system is fast, easy to configure, and does not require coding to generate event cycle reports. In fact, configuration can be accomplished declaratively by simply changing parameters associated with ALE Event Cycle connectors in the RFID Anywhere Administrator Console. The most important options include:

- ◆ **Report Frequency** The time between event cycle reports is configurable for each ALE Event Cycle connector. An assembly line

system may need very frequent updates, for example, while an inventory application may need only occasional reports.

- ◆ **Tag Filtering** Filtering protects networks and off-site applications from unwanted and unnecessary data. For example, if an application is only concerned with the arrival of goods from one supplier, then tags on goods from other suppliers can be excluded from event cycle reports. Or, if a shipment had both tagged pallets and tagged cases, RFID Anywhere could be configured to report only on pallet tags.
- ◆ **Tag Grouping** Grouping tag data at the edge makes back-end processing less difficult, reducing application complexity. For example, all of the pallets from a certain company could be listed together under a common group name.
- ◆ **Logical Readers** Physical readers are abstracted into logical reader groups by the RFID ALE Protocol in RFID Anywhere. This allows each ALE Event Cycle connector to specify one or more logical readers on which its reports are based. In this way, only data from specified readers will be included in reports, which is important for a system that connects to multiple pieces of hardware or that supports multiple applications.
- ◆ **Subscribers** This property contains a list of configured messaging connectors to which generated event cycle reports will be sent. This property can be left empty when defining the event cycle connector if some of the operational modes below will be used in an application.

☞ For information on defining event cycle connectors programmatically, or interacting with ALE directly from web services or business modules, see “Using ALE programmatically” [*RFID Anywhere Developer’s Guide*, page 53].

## Immediate operational mode

The **immediate operational mode** for RFID Anywhere’s ALE implementation is synchronous, and allows an application to define an ALE event cycle on the fly by providing RFID Anywhere with an XML file that defines the event cycle. This is done using the **Immediate(ECSpec)** method. RFID Anywhere then generates the report over a number of read cycles as defined, and sends the application back the generated event cycle reports.

The event cycle created in this way is not available for other applications to interact with or subscribe to, as it is meant to be an ad-hoc way to define and receive event cycle reports as necessary.

## Poll operational mode

The **poll operational mode** for RFID Anywhere’s ALE implementation is

also synchronous, and allows an application to request the output of a pre-configured event cycle. The pre-configured event cycle would have a name, and could have been defined either through the Administrator Console, or by previously using the **Define**(*name*, *ECSpec*) method. The method **Poll**(*name*) is used for the poll operational mode.

Using the poll operational mode allows an event cycle to be pre-configured in RFID Anywhere, but only executed when a report is desired. That event cycle connector would not keep track of data unless it had been tolled to using the poll method, or if it also had associated subscribers.

### Subscribe operational mode

The **subscribe operational mode** is asynchronous, and is used to dynamically add a subscriber to a pre-configured event cycle connector. The pre-configured event cycle would have a name, and could have been defined either through the Administrator Console, or by previously using the **Define**(*name*, *ECSpec*) method. To do this, applications use the method **Subscribe**(*name*, *destination*) where *name* is the name of the pre-configured event cycle connector, and *destination* is the name of a pre-configured messaging connector in RFID Anywhere.

After this method is called, RFID Anywhere's ALE implementation sends reports to the destination each time an event cycle completes.

### The ECSpec parameter

The ECSpec parameter, used by the Immediate and Define methods, is an XML representation of an event cycle connector.

The following is a sample ECSpec definition file.

```
<?xml version="1.0" encoding="UTF-8"?>
<ECSpec schemaVersion="1.0" creationDate="2003-08-
    06T10:54:06.444-05:00">
  <logicalReaders>
    <logicalReader>Source1</logicalReader>
  </logicalReaders>
  <boundarySpec>
    <startTrigger>http://sample.com/trigger1</startTrigger>
    <repeatPeriod unit="MS">2000</repeatPeriod>
    <stopTrigger>http://sample.com/trigger2</stopTrigger>
    <duration unit="MS">3000</duration>
  </boundarySpec>
```

```
<reportSpecs>
  <reportSpec reportName="report1">
    <reportSet set="CURRENT"/>
    <output includeList="true"/>
  </reportSpec>
  <reportSpec reportName="report2">
    <reportSet set="ADDITIONS"/>
    <output includeCount="true"/>
  </reportSpec>
  <reportSpec reportName="report3">
    <reportSet set="DELETIONS"/>
    <groupSpec>
      <pattern>urn:epc:pat:sgtin-96:X.X.X.*</pattern>
    </groupSpec>
    <output includeCount="true"/>
  </reportSpec>
</reportSpecs>
</ECSpec>
</ECReports>
```

## The ALE Web Service module

RFID Anywhere’s ALE Web Service module ALEService provides a secondary means of interacting with the ALE engine by exposing methods over the web using SOAP.

Since some ALE applications may need reports only at specific times, or only as a result of user action, developers can use the Web Service module to define and trigger one-time ALE reports or retrieve important parameters.

The methods exposed through the ALE Web Service enable the various operational modes discussed above.

☞ To view the methods available through RFID Anywhere’s ALE Web Service in the form of a WSDL file, click the **ALEService** link in the list of Local Services from the Home page of the Administrator Console.

☞ For more detailed information on the available ALE methods, see “Available ALE methods” [*RFID Anywhere Developer’s Guide*, page 54].

## Adding business context to ALE reports

Event cycle reports in the form of XML documents are generated within RFID Anywhere, and sent to enterprise applications through messaging connectors or through web services. Once an enterprise application receives an event cycle report, it is the job of that application to parse the report, and add business context to the underlying information. For example, the tag ID for each element in the report may need to be mapped to a specific part

number or manufacturer for display. With this information, the enterprise application can then act on that information.

## About this tutorial

This tutorial takes you through the steps of using ALE to filter and group the simulated data that you created in the previous chapter. You use the File, TCP, and MSMQ messaging connectors to send ALE reports to an enterprise application.

It is important to note that ALE filters and groups the data, but does not apply any additional business logic. When a report is generated and sent to an enterprise application over one of RFID Anywhere's messaging connectors, it is up to the application to add additional business value and context to the data contained in the report.

**Note**

This tutorial assumes that you installed the ALE Report Viewer when you installed RFID Anywhere.

Tasks you will complete in this tutorial include:

- ◆ Adding connectors to the Local Services list
- ◆ Configuring the RFID Multiprotocol Simulator connector
- ◆ Configuring the File messaging connector
- ◆ Configuring the Tcp messaging connector
- ◆ Configuring the MSMQ messaging connector
- ◆ Configuring the ALE Event Cycle connector
- ◆ Configuring the RFID ALE Protocol
- ◆ Viewing and understanding the output

# Lesson 1: Working with connectors from the Administrator Console

This tutorial uses a number of the connectors included with RFID Anywhere. Note that the RFID Multiprotocol Simulator connector and the Tcp messaging connector that you configure in this lesson will be used for the Inventory Tracker demo. Thus the naming conventions for these connectors will reflect the settings required for Inventory Tracker.

## Adding the RFID Multiprotocol Simulator connector

The RFID Multiprotocol Simulator connector simulates the appearance of the tags that were defined using the RFID Simulator Data Editor.

### Note

If you have not already created the **InvTracSim1** RFID Multiprotocol Simulator connector, see [“Lesson 3: Configuring the RFID Multiprotocol Simulator connector” on page 150](#) for instructions on creating and configuring the connector.

## Adding the File messaging connector

The File messaging connector allows processed information in the form of files to be saved to the file system. When configured to save files in `.txt` format, a generic text editor can be used to view the data.

☞ For detailed information on File messaging and all of the other messaging connectors, see [“RFID Anywhere Messaging Connectors” on page 101](#).

### ❖ To add and configure the File messaging connector

1. In the Administrator Console, choose File Messaging from the Service Type dropdown list and then click Add.

The File messaging connector is added to the list of Local Services and is highlighted. The new File messaging connector has the default name `FileConnector1`.

2. Edit the properties of the new connector by selecting the checkbox beside it and then clicking Edit.

The properties for the connector appear in the Properties Manager pane on the right side of the window.

3. In the textbox for the Name property, type **FileOutput**.

4. In the textbox for the Directory property, enter **C:\RFID Output** to specify the folder where the files from the simulation output is stored. This directory is created on your computer (if it does not already exist) when you start the connector.
5. Click the green disk in the Properties Manager title bar to save the above changes.

The File messaging connector is updated.

## Adding the Tcp messaging connector

The Tcp messaging connector is used to send processed information from RFID Anywhere via the TCP communications protocol. The output can be received by any external application that is listening on the configured port. In this tutorial, you will use the ALE Report Viewer application to receive the output.

As mentioned earlier, the Tcp messaging connector configured in this section is also used in upcoming chapters when setting up the Inventory Tracker demo.

### ❖ To add and configure the Tcp messaging connector

1. Choose Tcp Messaging from the Service Type dropdown list and then click Add.

The Tcp messaging connector is added to the list of Local Services and is highlighted. The new Tcp messaging connector has the default name TcpIpConnector1.

2. Edit the properties of the new connector by selecting the checkbox beside it and then clicking Edit.

The properties for the connector appear on the right side of the window.

3. In the textbox for the IP Address property, type **localhost**.
4. In the textbox for the Name property, type **InvTracAleTCP**.
5. In the textbox for the Port property, type **10002**.
6. Click the green disk in the Properties Manager title bar to save the above changes.

The Tcp messaging connector is updated.

## Adding the MSMQ messaging connector

The MSMQ messaging connector is used to transmit RFID Anywhere processed information to applications via the MSMQ communications

protocol. The output can be received by applications that can pick up messages from the configured queue. In this tutorial, you will use the ALE Report Viewer application to receive the output.

❖ **To add and configure the MSMQ messaging connector**

1. Choose MSMQ Messaging from the Service Type dropdown list, and then click Add.

The MSMQ messaging connector is added to the list of Local Services and is highlighted. The new MSMQ messaging connector has the default name MSMQConnector1.

2. Edit the properties of the new connector by selecting the checkbox beside it and then clicking Edit.

The properties for the connector appear on the right side of the window.

3. In the textbox for the Computer Name property, type **localhost**.

4. In the textbox for the Name property, type **MSMQOutput**.

5. In the textbox for the Private Queue Name property, type **reader1Output**.

6. Click the green disk in the Properties Manager title bar to save the above changes.

The MSMQ messaging connector is updated.

## Adding the ALE Event Cycle connector

The ALE Event Cycle connector adheres to the EPC standard and specifies the correct format in which to send processed information (reports). The connector can be configured to filter and group certain types of tags so that only relevant information is transported across the network. For example, the connector can be configured to filter out products from certain manufacturers.

In this tutorial, the ALE Event Cycle connector receives simulated data (either *reader1manual.smt* or *reader1.smt*) from the InvTracSim1 RFID MP Simulator connector. The ALE Event Cycle connector will be configured to filter the raw RFID data and report only on AutoClean 77 Sterilizers from Robertson Health. There are three such sterilizers in the simulation data and their tag IDs are 1.1.2.5, 1.1.2.6, and 1.1.2.7, respectively.

☞ For more information on the tags discussed in this chapter, see [“Creating Simulated RFID Data” on page 137](#) and the file *TagActivitiesTimeline\_Medical.xls* located in the *Inventory*

*Tracker\Documentation\Optional* folder of your RFID Anywhere installation directory.

❖ **To add and configure the ALE Event Cycle connector**

1. Choose ALE Event Cycle from the Service Type dropdown list, and then click Add.

A new ALE Event Cycle connector appears in the Local Services list. The new ALE Event Cycle connector has the default name AleECCconnector1.

2. Select the new ALE Event Cycle connector and then click Edit. The Properties Manager for the ALE Event Cycle connector appears to the right.
3. Expand the Event Cycle set of properties, then expand the Start stop conditions set of properties.
4. In the text field for the Duration (ms) property, type **1000**.
5. Expand the Logical Readers property. In the text field, type **Source1** and then click the green plus sign to add the entry.
6. In the textbox for the Report(s) Definition property, type **1**.
7. Expand the reportSpecs(1) set of properties. In the textbox for the Include Patterns property, enter **urn:epc:pat:sgtin-96:1.1.2.\***.

The Include Patterns property is used to specify the tags that you want to be included in the report. When a value is entered for this property, only the tags that are specified will be reported. For this tutorial, you only want the AutoClean 77 Sterilizer tags.

The Exclude Patterns property is used to specify the tags that you want to be excluded from the reports.

The Grouping Patterns property can be used to group certain tags together. For example, tags can be grouped according to their manufacturers.

☞ For a list of all grouping and filtering patterns, see [“Using ALE to Filter and View RFID Data” on page 153](#).

8. In the dropdown list for the reportSpecs(1) ► Report If Empty property, choose True.

When this property is set to True, the ALE Event Cycle connector sends reports, even if it does not contain any tags, so the ALE Event Cycle connector sends empty reports when there are no tags in the proximity of the reader antennas.

9. In the textbox for the reportSpecs(1) ► Report Name property, type **Source1Filtered**.
10. In the dropdown list for the reportSpecs(1) ► EPC to be reported property, choose CURRENT.
11. In the textbox for the reportSpecs(1) ► Name property, type **FilteredSource1ALE**.
12. In the dropdown list for the Send Reports as Object property, choose False.
13. Expand the Subscribers property. In the text box, enter **FileOutput** and press the green plus sign to add this item to the Subscribers list. Follow the same steps to add **InvTracAleTCP** and **MSMQOutput** to the Subscribers list.
14. Click the green disk in the Properties Manager title bar to save the above changes.

The ALE Event Cycle connector is updated.

## The RFID ALE Protocol business module

The RFID ALE Protocol business module provides the business logic to link the reader group definition to the reader that is gathering data (RFID MultiProtocol Simulator).

The RFID ALE Protocol business module is automatically installed in the list of Local Services and has the name ALEServiceModule.

### ❖ To configure the RFID ALE Protocol business module

1. Edit the properties of the RFID ALE protocol by selecting the checkbox beside ALEServiceModule, and then clicking Edit.

The properties for the business module appear on the right side of the window.

2. In the textbox for the Logical Readers property, type **Source1:InvTracSim1**.

This specifies that the RFID ALE Protocol business module should use the Source1 group of readers and connect to the InvTracSim1 RFID Multiprotocol Simulator connector as the raw data source. If there were other groups of readers in the network, another entry could be added to the properties (for example, Source2:InvTracSim2).

3. Click the green disk in the Properties Manager title bar to save the above changes.

The RFID ALE Protocol business module is updated.

## Filtering and grouping patterns

**Note**

This section may be skipped without loss of continuation.

The following table lists the syntax that can be used for ALE filtering. These values can be entered in the Exclude Patterns and Include Patterns properties of an ALE Event Cycle connector.

To create these filters, you need to understand the contents and format of the specific EPC encoding being used. For example, to create the filters below, you need to know that the Serialized Global Trade Item Number 96-bit encoding (SGTIN-96) consists of 4 fields—a Filter, Company Prefix, Item Reference, and a Serial Number.

ALE filtering

Pattern URI (syntax)	Description
urn:epc:pat:sgtin-96:1.678.5.3470	Matches the tag for the specific item that has a Filter value of 1, Company Prefix of 678, an Item Reference of 5, and a Serial Number of 3470. A filter such as this might be useful for an application that is designed to send a notification when a specific product instance from a company arrives.
urn:epc:pat:sgtin-96:1.678.5.*	Matches all tags of a specific Filter (1), Company Prefix (678) that have an Item Reference of 5, regardless of Serial Number. A filter such as this would be useful for filtering out all assets of a certain type.
urn:epc:pat:sgtin-96:1.678.*.[5000-9999]	Matches any tag that has a Filter value of 1, a Company Prefix of 678 and a serial number between 5000 and 9999 inclusive, regardless of the Item Reference. A filter such as this would be useful for filtering out certain instances of all types of products arrived from a specific manufacturer.
urn:epc:pat:sgtin-96:*.*.*.*	Matches any SGTIN-96 tag.

The following table lists the syntax that can be used for ALE grouping. These values can be entered in the Grouping Patterns property of an ALE Event Cycle connector. The last syntax combines both grouping and filtering.

Again, to create these groupings, you need to understand the contents and format of the specific EPC encoding being used.

### ALE grouping

Pattern URI (syntax)	Description
urn:epc:pat:sgtin-96:X.X.*.*	Groups tags by Filter value and Company Prefix (for example, groups according to manufacturers).
urn:epc:pat:sgtin-96:X.*.X.*	Groups tags by Filter value and Item Reference (for example, groups according to specific product types).
urn:epc:pat:sgtin-96:X.X.*.[0-100]	Creates a different group for each Filter value and Company Prefix, including in each group only tags that have Serial Numbers in the range 0 through 100, inclusive.

## Lesson 2: Starting the connectors

You must start the connectors created in Lesson 1 to begin the simulation. Connectors are started from the Administrator Console.

### ❖ To start the connectors

1. Select the checkboxes beside the following connectors, and then click Start:

- ◆ FilteredSource1ALE
- ◆ FileOutput
- ◆ InvTracSim1
- ◆ MSMQOutput
- ◆ InvTracAleTCP

The connectors are identified with green arrows once they start. If an error message appears above the list of Local Services, ensure that you have followed the steps correctly and have not made any spelling errors.

## Lesson 3: Viewing and understanding ALE reports

This lesson outlines two options for viewing generated ALE reports. It is useful to view and understand the reports generated by event cycle connectors when designing or troubleshooting an ALE-based application.

### Using the File messaging connector to receive reports

The File messaging connector is used to send processed information (reports) in the form of files to a user-specified location in the file system. It can be configured to save the files in any format, such as the `.txt` format. When configured to send files in `.txt` format (such as in FileOutput), one advantage of the File messaging connector is that the files can be easily accessed and viewed using a generic text editor.

#### ❖ To view the processed information via a text editor

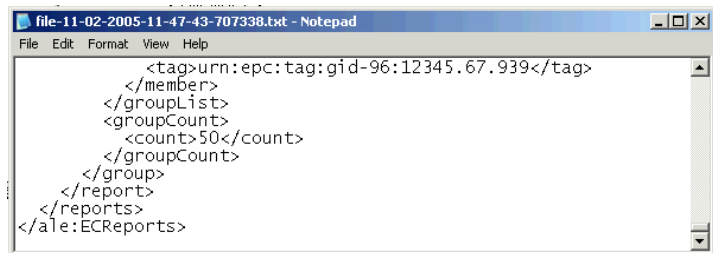
1. In the Administrator Console, if not already started, select FilteredSource1ALE, FileOutput, and InvTracSim1, and then click Start.
2. Browse to `C:\RFID Output`.

You configured the File Messaging Connector to save reports in this folder when you set the connector's properties. Since you specified the Duration property as 1000(ms) in the ALE Event Cycle connector, a new text file is created in this folder every second. Each text file contains the tags (if any) that were present during the previous second. The name of each text file is in the format `file-dd-mm-yyyy-hh-mm-ss-number.txt`.

3. Using a text editor, open the first file that appeared.

The file appears in the format specified by EPCGlobal, Inc. (<http://www.epcglobalinc.org>) and includes all the tags that have appeared since the last cycle.

The number of tags that appear in each report is reported in the `<groupCount>` element near the end of the report file.



```

file-11-02-2005-11-47-43-707338.txt - Notepad
File Edit Format View Help
<tag>urn:epc:tag:gid-96:12345.67.939</tag>
</member>
</groupList>
<groupCount>
<count>50</count>
</groupCount>
</group>
</report>
</reports>
</ale:ECReports>

```

## The ALE Report Viewer application

The RFID Anywhere ALE Report Viewer is an application that demonstrates how an external application can receive and process ALE reports. The ALE Report Viewer can be configured to receive ALE reports from four types of messaging connectors: QAnywhere, MSMQ, File Messaging, and TCP.

The RFID Anywhere ALE Report Viewer receives ALE reports from messaging connectors, and then simply displays the reports in a convenient format. The display includes important information contained in the reports such as the source name, report name, time reported, number of groups, number of items, and the size of the reports.

### Using File Monitoring to receive reports

The File Monitoring section of the ALE Report Viewer lets you connect to a file messaging connector and receive ALE reports.

#### ❖ To receive processed information using the File messaging connector

1. In the Administrator Console, if not already started, select FilteredSource1ALE, FileOutput, and InvTracSim1, and then click Start.
2. In your RFID Anywhere installation directory, navigate to the *bin* folder. By default, this directory is located in *C:\Program Files\Sybase\RFID Anywhere\bin*.
3. Double-click *iAnywhere.RfidNet.Ale.ReportViewer.exe*.  
The RFID Anywhere ALE Report Viewer opens.
4. Click File Monitoring.  
From here, you can connect to the file messaging connector to receive ALE reports.
5. Enter the following information:

◆ **Directory Path** C:\RFID Output

This is the same path that was specified for the FileOutput file messaging connector in the Administrator Console.

◆ **Wildcard** \*.txt

Since the FileOutput File messaging connector sends files in *.txt* format, the Report Viewer must be configured to accept input in *.txt* format.

**Note**

You can change the output format of FileOutput (the file messaging connector) from the Properties Manager in the Administrator Console, but you must always ensure that the format specified in the Report Viewer (under Wildcard) and the output format of FileOutput are the same.

6. In the RFID Anywhere ALE Report Viewer, click Start on the File Monitoring page.  
The application starts receiving reports.
7. Once you have received some reports, click Stop.
8. In the Administrator Console, select FilteredSource1ALE, FileOutput, and InvTracSim1, and then click Stop.  
The ALE event cycle, the file messaging connector, and the RFID MP simulator connectors stop.

### Using TCP/IP to receive reports

The TCP/IP tab of the RFID Anywhere ALE Report Viewer lets you connect to a Tcp messaging connector and receive ALE reports.

#### ❖ To receive processed information using the Tcp messaging connector

1. Select FilteredSource1ALE, InvTracAleTCP, and InvTracSim1, and then click Restart.  
The ALE event cycle, the Tcp messaging connector, and the RFID MultiProtocol simulator connectors restart.
2. Open the RFID Anywhere ALE Report Viewer if it is not already open.
3. Click the TCP/IP tab.  
The TCP/IP pane appears. From here, you can connect to the TCP/IP messaging connector to receive ALE reports.
4. Set the value for Server Port to **10002**.  
The value for Server Port should be the same as the Port property of InvTracAleTCP.
5. Click Clear if there is output already in the RFID Anywhere ALE Report Viewer display.
6. Click Start to start the receiving process.  
The processed information appear in the form of reports.

7. Once the reports are being received, click Stop.

In the Administrator Console, select FilteredSource1ALE, InvTracAleTCP, and InvTracSim1, and then click Stop.

The ALE event cycle, the Tcp messaging connector, and the RFID MP simulator connectors stop.

## Using MSMQ to receive reports

This section shows you how to use the ALE Report Viewer to connect to an MSMQ messaging connector and receive ALE reports.

### ❖ To receive processed information using the MSMQ messaging connector

1. Select FilteredSource1ALE, MSMQOutput, and InvTracSim1, and then click Start.

The ALE event cycle, MSMQ messaging connector, and the RFID MP simulator connectors start.

2. Open RFID Anywhere ALE Report Viewer if it is not already open.

3. Click the MSMQ tab.

The MSMQ pane appears. From here you can connect to the MSMQ messaging connector to receive ALE reports.

4. Ensure that the Queue Name is set to **reader1Output**.

The value for Queue Name should be the same as the Private Queue Name property of MSMQOutput.

5. Click Clear if there is output already in the RFID Anywhere ALE Report Viewer display.

6. Click Start to start the receiving process.

The processed information appear in the form of reports.

7. Once reports are being received, click Stop.

8. In the Administrator Console, select FilteredSource1ALE, MSMQOutput, and InvTracSim1, and then click Stop.

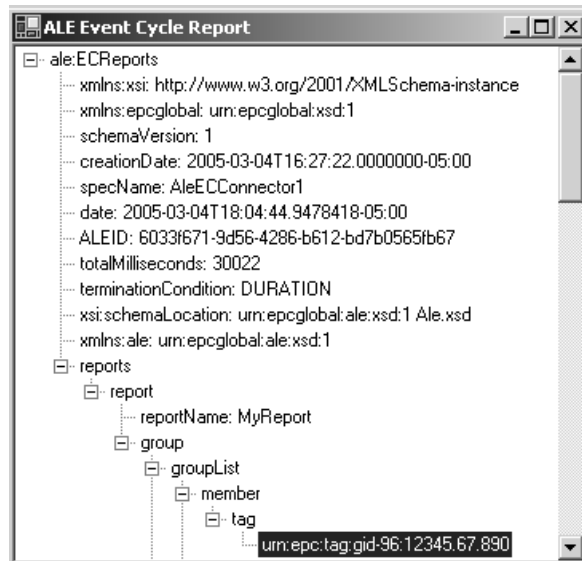
The ALE event cycle, the MSMQ messaging connector, and the RFID MP simulator connectors stop.

## Understanding the output

Typical output in the RFID Anywhere ALE Report Viewer looks similar to the following:

ID	Source	Reports	Event Cycle	Time	Groups	Items	Size
10	TCP	MyReport	AleECConne	6:03:44 PM	1	50	16704
11	TCP	MyReport	AleECConne	6:04:45 PM	1	50	5568
12	TCP	MyReport	AleECConne	6:05:15 PM	1	25	3168
13	TCP	MyReport	AleECConne	6:05:45 PM	1	50	5568
14	TCP	MyReport	AleECConne	6:06:15 PM	1	25	3168
15	TCP	MyReport	AleECConne	6:06:45 PM	1	50	5568
16	TCP	MyReport	AleECConne	6:07:15 PM	1	25	3168
17	TCP	MyReport	AleECConne	6:07:45 PM	1	50	5568
18	TCP	MyReport	AleECConne	6:08:15 PM	1	50	5568
19	TCP	MyReport	AleECConne	6:09:15 PM	1	50	5568
20	TCP	MyReport	AleECConne	6:09:45 PM	1	1	815
21	TCP	MyReport	AleECConne	6:10:15 PM	1	50	5568
22	TCP	MyReport	AleECConne	6:10:45 PM	1	25	3168
23	TCP	MyReport	AleECConne	6:11:15 PM	1	50	5568
24	TCP	MyReport	AleECConne	6:11:46 PM	1	25	3168
25	TCP	MyReport	AleECConne	6:12:16 PM	1	50	5568
26	TCP	MyReport	AleECConne	6:12:47 PM	1	26	3265
27	TCP	MyReport	AleECConne	6:13:17 PM	1	50	5568

The Reports list shows some of the properties for the ALE reports that have been received. The displayed properties include important information contained in the reports such as the source name, report name, time reported, number of groups, number of items, and the size of each of the reports. You can double-click a report to see more detailed information. Navigate through the output by either clicking + to expand the content or – to close it.



These reports clearly demonstrate RFID Anywhere's ability to filter information at the edge of an RFID Anywhere network. Moreover, since all three messaging connectors (FileOutput, InvTracAleTCP, and MSMQOutput) transmit processed information generated from the same source, it showcases RFID Anywhere's ability to allow multiple enterprise applications to share the same data.

## Lesson 4: Stopping the connectors

The connectors must be stopped in order to end the simulation. They can be stopped from the Administrator Console.

### ❖ To stop the simulation

1. Select the checkbox beside FilteredSource1ALE, FileOutput, InvTracAleTCP, MSMQOutput, and InvTracSim1, and then click Stop. A red X appears beside the name of each connector once they have stopped.

## Properties of ALE Event Cycle connectors

The specific properties for defining an ALE Event Cycle report, including the reader connectors to get raw data from, filtering, the report duration, and the RFID Anywhere messaging transport that will deliver the report are configured declaratively from the RFID Anywhere Administrator Console. The following properties can be configured when defining an ALE Event Cycle connector from the RFID Anywhere Administrator Console.

Property	Description
EC Creation Date	Shows the date and time when the event cycle was created.
Description	Provides a description of the connector that appears in the Local Services list.
Duration	Specifies the amount of time the event cycle collects data for.
Repeat Period	Specifies the interval an event cycle waits before it starts the next event cycle. When set to <b>0</b> , there is no delay between the end of one event cycle and the beginning of the next one.
Stable Interval	Specifies the interval that determines if an event cycle is stable and should finish because no new tags have been reported.
Start Trigger	Starts a defined event cycle. You can trigger this property from another business module or set the value in the Administrator Console.
Stop Trigger	Stops a defined event cycle.
Report EC Spec	If this value is set to True, then in the case of a standing report request, reports are not sent to subscribers unless the filtered list of EPCs is different than the previously-filtered list of EPCs. When set to False, reports are sent to subscribers regardless of whether the filtered list of EPCs is different than the previously-filtered list.
Logical Readers	Defines the groups of logical readers to collect data from. You can only specify group names. The relationship between the group names and the associated readers are specified using the ALE service module.

Property	Description
Report Definition	Specifies the report to be returned from executing an event cycle.
Exclude Patterns	<p>Specifies a list of excluded EPC patterns.</p> <ul style="list-style-type: none"> <li>◆ <b>urn:epc:pat:gid-96:678.9012.3470</b> Matches the EPC for a particular product from manufacturer B with serial number 3470.</li> <li>◆ <b>urn:epc:pat:gid-96:678.9012.*</b> Matches EPC of particular family of products (example electronics) from manufacturer B, regardless of serial number.</li> <li>◆ <b>urn:epc:pat:gid-96:12345.*[5000-9999]</b> Matches any manufacturer, A, product whose serial number is between 5000 and 9999, inclusive.</li> <li>◆ <b>urn:epc:pat:gid-96:*.*.*</b> Matches any GID-96 tag.</li> </ul>
Grouping Patterns	<p>Defines how filtered EPCs are grouped together for reporting Each element of the pattern list is an EPC Pattern URI as defined by the EPCglobal Tag Data Specification Version 1.1 [TDS1.1], extended by allowing the character X in each position where a * character is allowed.</p> <ul style="list-style-type: none"> <li>◆ <b>urn:epc:pat:gid-96:X.*.*</b> Groups by company prefix.</li> <li>◆ <b>urn:epc:pat:gid-96:X.X.*</b> Groups by company prefix and item reference (for example, groups by specific product).</li> <li>◆ <b>urn:epc:pat:gid-96:X.X.*</b> Groups by company prefix and item reference.</li> <li>◆ <b>urn:epc:pat:gid-96:X.*[0-100]</b> Creates a different group for each company prefix, including in each such group only EPCs that have serial numbers in the range 0 through 100, inclusive.</li> </ul>
Include Count	If includeCount is True, the report includes a count of the EPCs in the final set for each group.

Property	Description
Include Pure Identity URI	The EPC field is an EPC represented as a pure identity URI according to the EPCglobal Tag Data Specification (urn:epc:id:...).
Include Patterns	Lists the included EPC patterns.
Include Raw Decimal	The rawDecimal field is a raw tag value represented as a raw decimal URI according to the EPCglobal Tag Data Specification (urn:epc:raw:...).
Include Raw Hex	The rawHex field is a raw tag value represented as a raw hexadecimal URI according to the following extension to the EPCglobal Tag Data Specification: the URI shall be determined by concatenating the following: the string urn:epc:raw:, the length of the tag value in bits, a dot (.) character, a lowercase x character, and the tag value considered as a single hexadecimal integer.
Include Tag URI	The tag field is an EPC represented as a tag URI according to the EPCglobal Tag Data Specification (urn:epc:tag:...).
Report if Empty	Defines whether an empty report should be sent to subscribers.
Report Name	Identifies that ECRReportSpec must be unique in the ECSpec.
Report on Change	If an ECRReportSpec has reportOnlyOnChange set to true, in the case of a standing report request, then reports are not sent to subscribers unless the filtered list of EPCs is different than the previously filtered list of EPCs.
EPC to be Reported	Shows whether an event cycle has been requested. This property must have one of the following values: <ul style="list-style-type: none"> <li>◆ <b>CURRENT</b> Reports all EPCs read in the current event cycle.</li> <li>◆ <b>ADDITIONS</b> Reports only additions from the previous event cycle.</li> <li>◆ <b>DELETIONS</b> Reports only deletions from the previous event cycle.</li> </ul>

Property	Description
Encoding Requested	The encoding used to send reports to subscribers.
Name	The connector name that appears in the Local Services list. This is the name you use to refer to this connector from other local services.
Send Reports as Object	True sends the result as an ECRReport object serialized to XML, while false sends an XML string.
Start Mode	<p>A value indicating how the service starts.</p> <ul style="list-style-type: none"> <li>◆ <b>Automatic</b> The service is started by default when the Administrator Console is loaded and/or when the service is added to the Local Services list.</li> <li>◆ <b>Manual</b> The service is turned off by default. It must be started by the user via the Administrator Console. This is the default setting.</li> </ul>
Start Operation	<p>If the Start Mode property of the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Configure method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Subscribers	Defines the messaging transports that receive and transmit event cycle reports.
Synchronous Event Cycle Timeout	Defines the timeout in milliseconds for immediate ALE method calls.

Property	Description
Trace Level	<p>This property specifies the level of information written to the log file. The value can be one of the following:</p> <ul style="list-style-type: none"><li>◆ <b>Off</b> No messages are written to the log file.</li><li>◆ <b>Error</b> Only error messages are written to the log file. This is the default value.</li><li>◆ <b>Warning</b> Only warning messages are written to the log file.</li><li>◆ <b>Info</b> Only informational messages are written to the log file.</li><li>◆ <b>Verbose</b> All messages are written to the log file.</li></ul>

## Summary

You now have knowledge of what Application Level Events (ALE) are and how you can generate filtered data with them. ALE filters and groups the data, but does not apply any additional business logic. It is up to the enterprise application that receives the reports to add business value.

The ALE engine is used to generate processed information (XML based reports) that adhere to the ALE specifications published by EPCGlobal, Inc. (<http://www.epcglobalinc.org>).

ALE can only be used to generate reports on EPC tags, and not on any other tag protocols such as ISO. With RFID Anywhere you are not limited to just EPC tags. The next chapter introduces the RFID Report Engine MP, which can be used to generate XML reports on any tag protocol.



---

## CHAPTER 11

# Using Report Engine MP

About this chapter

This chapter introduces the Report Engine MP and shows how it can be used to generate processed information in the form of XML reports.

Contents

<b>Topic:</b>	<b>page</b>
<a href="#">Introduction</a>	184
<a href="#">Lesson 1: Working with connectors from the Administrator Console</a>	186
<a href="#">Lesson 2: Starting the demo</a>	189
<a href="#">Lesson 3: Viewing the simulation reports</a>	190
<a href="#">A sample multiprotocol report</a>	191
<a href="#">Lesson 4: Stopping the connectors</a>	193
<a href="#">Properties of RFID MultiProtocol Report Definition connectors</a>	194
<a href="#">Summary</a>	201

## Introduction

The Report Engine MP feature of RFID Anywhere enables the simple creation of XML reports based on EPC, ISO, or any other tag protocol.

While the ALE standard enables the generation of XML reports for tag activity, it is limited in that it can only generate these reports for EPC tags. This limitation is addressed with RFID Anywhere's Report Engine MP by enabling the creation of XML reports based on tag activity for both EPC and ISO tags. The Report Engine MP utilizes RFID Anywhere's Data Protocol Processor (DPP) functionality to decode the tags of various protocols for the creation of consolidated reports.

One of the other advantages of Report Engine MP over ALE is that it explicitly states various parts of raw tag IDs. For example, if the tags are in the SGTIN-96 EPC format, Report Engine MP XML reports will list the Filter, Company Prefix, Product Name, and the Serial Number as separate attributes.

Report Engine MP filters and groups data, but does not apply business logic. After a report is generated and sent to an enterprise application, it is the job of that application to process the report and apply business logic. This is similar to how ALE reports are used.

☞ For more information on the Data Protocol Processor (DPP), see [“Getting Started with Data Protocol Processors \(DPPs\)” on page 95](#).

## Building RFID solutions using Report Engine MP

Applications built with Report Engine MP are driven by XML reports that summarize tag activity over a period of time.

The Report Engine MP development option consists of several key components: the hardware connectors, which communicate with the readers, the pre-installed RFID Multiprotocol Service Module, which collects raw data from the hardware controller for RFID readers, the RFID MultiProtocol Report Definition connector, which defines and validates reports, and messaging transports, which deliver the completed reports to enterprise applications for processing.

The RFID Anywhere Report Engine MP system is fast, easy to configure, and does not require coding to generate XML reports that summarize tag activity. In fact, configuration can be accomplished declaratively by simply changing parameters associated with the RFID MultiProtocol Report Definition connectors in the RFID Anywhere Administrator Console.

## About this tutorial

This tutorial will take you through the steps of using Report Engine MP to filter/group the simulated data that you had created earlier. You will use the File messaging connector to send reports to the file system.

It is important to note that Report Engine MP filters and groups the data, but does not apply any business logic. When a report is generated and sent to an enterprise application over one of RFID Anywhere's messaging connectors, it is up to the application to add additional business value to the data contained in the report.

Tasks you will complete in this tutorial include:

- ◆ Adding connectors to the Local Services list
- ◆ Configuring the RFID Multiprotocol Simulator connector
- ◆ Configuring the File Messaging connector
- ◆ Configuring the RFID MultiProtocol Report Definition connector
- ◆ Configuring the Multiprotocol Service Module
- ◆ Viewing the simulation reports
- ◆ Properties of RFID Multiprotocol Report Definition connectors

# Lesson 1: Working with connectors from the Administrator Console

This tutorial makes use of multiple RFID Anywhere connectors. These include a File messaging connector, a MultiProtocol Report Definition connector, and a RFID Multiprotocol Simulator connector.

## RFID Multiprotocol Simulator connector

The RFID Multiprotocol Simulator connector simulates the appearance of the tags that were defined using the RFID Simulator Data Editor.

**Note**

If you have not already created the InvTracSim1 RFID Multiprotocol Simulator connector, see [“Creating Simulated RFID Data” on page 137](#) for instructions on creating and configuring the connector.

## The File messaging connector

The File Messaging connector allows processed information in the form of files to be saved to the file system. This connector also allows the processed information from RFID Anywhere to be easily viewed.

**Note**

If you did not already create the FileOutput File messaging connector in the previous tutorial, see [“Adding the File messaging connector” on page 160](#) for instructions on creating the connector.

## The RFID MultiProtocol Report Definition connector

The RFID MultiProtocol Report Definition connector performs similar functions to the ALE Event Cycle connector described in [“Using ALE to Filter and View RFID Data” on page 153](#). The RFID MultiProtocol Report Definition connector contains specific properties for defining a multiprotocol report, including the reader connectors to get raw data from, the type of tags to include (filtering), the report duration and the RFID Anywhere messaging transport(s) that will deliver the reports. The connector will generate reports based on these property values.

Similar to the ALE tutorial, you will again filter out the raw RFID data before sending the reports. For the purposes of this tutorial, you will filter the data so that only products from Robertson Health are included in the reports. There are eight products shipped from Robertson Health in the

simulation data and the their tag IDs all have a Company Prefix of 1.

☞ For details on all of the properties available when configuring an RFID MultiProtocol Report Definition connector, see [“Properties of RFID MultiProtocol Report Definition connectors”](#) on page 194.

❖ **To add and configure a RFID MultiProtocol Report Definition connector**

1. Open the Administrator Console.
2. Choose MultiProtocol Report Definition from the Service Type dropdown list, and then click Add.  
The RFID MultiProtocol Report Definition connector is added to the list of Local Services and is highlighted. The new connector has the default name MPReport1.
3. Edit the properties of the new connector by selecting the checkbox beside it and then clicking Edit.  
The properties for the connector appear on the right side of the window.
4. In the dropdown list for the Collection Type property, choose CURRENT.
5. Expand the DPP property. In the text field, type **RFIDTagEPC\_Class0:RA\_DPP** and press the green plus sign to add the entry to the list.  
This value must be the same as the value used in the RFID Simulator Data Editor when creating the tags. This property is case sensitive.
6. In the textbox for the Duration property, type **10s** to generate a report every 10 seconds.
7. In the dropdown list for the Return Table Layout property, choose False.  
Setting this property to True ensures that the underlying table definition of the data is included in the generated XML report within a `<xrnet:dataLayout>` element. When set to False, the table layout is not included in the generated report.
8. In the dropdown list for the Include SQL in Report property, choose True.
9. Expand the Logical Readers property. In the text field, type **MPSource** and press the green plus sign to add the entry to the list.
10. In the textbox for the Name property, type **MPReport**.

11. Expand the Queries property. In the text field, type **SELECT \* FROM sgtin96 WHERE CompanyPrefix = 1** and press the green plus sign to add the entry to the list.

This property is used to filter and group the tag data to be included in the report. The value entered for this property are SQL queries. As mentioned earlier, you want to filter the data such that you only include products from Robertson Health in the report. All products shipped from Robertson Health have a Company Prefix of 1.

To create multiple groups in one report, enter multiple SQL queries in this property by adding new entries to the Queries property.

☞ For more information on the table and column names used by the default RFID Anywhere DPP, see [“Using the default RFID Anywhere DPP for EPC tags” \[RFID Anywhere Developer’s Guide, page 88\]](#).

12. Expand the Subscribers property. In the text field, type **FileOutput** and press the green plus sign to add the entry to the list.
13. Click the green disk in the Properties Manager title bar to save the above changes.

The RFID MultiProtocol Report Definition connector is updated.

## The RFID Multiprotocol Service Module

The RFID Multiprotocol Service Module provides the business logic to link the reader group definition to the reader that is gathering data (RFID MultiProtocol Simulator).

The RFID Multiprotocol Service Module is automatically installed in the list of Local Services and has the name MPRServiceModule.

### ❖ To configure the RFID Multiprotocol Service Module

1. Edit the properties of the RFID Multiprotocol Service Module by selecting the checkbox beside it and then clicking Edit.

The properties for the connector appear on the right side of the window.

2. Expand the Logical Readers property. In the text field, type **MPSource:InvTracSim1** and press the green plus sign to add the entry to the list.

3. Click the green disk in the Properties Manager title bar to save the above changes.

The RFID Multiprotocol Service Module is updated.

## Lesson 2: Starting the demo

You must start the connectors created in Lesson 1 to begin the simulation. Connectors are started from the Administrator Console.

### ❖ To start the demo

1. Select the checkbox beside MPRServiceModule, MPReport, FileOutput, and InvTracSim1, and then click Start.

The connectors are identified with green arrows once they start. If an error message appears above the list of Local Services, ensure that you have followed the steps correctly and have not made any spelling errors.

## Lesson 3: Viewing the simulation reports

As stated above, the File messaging connector is used to send processed information (reports) in the form of files to a user-specified location in the file system. It can be configured to save the files in any format, such as the *.txt* format. When configured to send files in *.txt* format (as you did when configuring the FileOutput connector), one advantage of the File messaging connector is that the files can be easily viewed using a generic text editor.

### ❖ To view the processed information via a text editor

1. In the Administrator Console, if not already started, select MPRServiceModule, MPReport, FileOutput, and InvTracSim1, and then click Start.

2. Browse to *C:\RFID Output*.

You configured the File Messaging Connector to save reports in this folder when you set the connector's properties. Since in the RFID MultiProtocol Report Definition connector, you specified the Duration property as 10s, a new text file is created in this folder every 10 seconds. Each text file contains the tags (if any) that were present during the previous second. The name of each text file is in the format *file-dd-mm-yyyy-hh-mm-ss-number.txt*.

3. Using a text editor, open the first file that appeared.

The file includes the filtered tags that have appeared since the last cycle.

Note that the reports contain various attributes of the tags. Since your simulation tags are in the EPC SGTIN-96 format, for each tag, the reports list the Filter, Company Prefix, Product Name, and the Serial Number as separate attributes.

## A sample multiprotocol report

After a report is generated and sent to an enterprise application, it is the job of that application to process the report and act on the information. The XML document must be parsed to extract the valuable information inside.

Below is a sample XML-based multiprotocol report generated by RFID Anywhere's Report Engine MP. The connector used to generate this report had the following properties:

- ◆ **Collection Type** CURRENT
- ◆ **DPP** RFIDTagEPC\_Class0:RA\_DPP
- ◆ **Duration** 10s
- ◆ **Return Table Layout** False
- ◆ **Include SQL in Report** True
- ◆ **Logical readers** MPSource (which maps to the Logical reader name defined in the MPRServiceModule)
- ◆ **Queries** SELECT \* FROM sgtin96 WHERE CompanyPrefix = 1
- ◆ **Subscribers** FileOutput (where FileOutput is the File Messaging connector configured to save the XML report in a folder on the local computer)

The MPRServiceModule was configured with the following properties:

- ◆ **Logical readers** MPSource:InvTracSim1 (which allows report connectors to enter MPSource in their Logical readers property to receive raw data from the hardware connector called InvTracSim1)

The XML report generated from these properties has the following contents:

```

<?xml version="1.0" encoding="utf-8"?>
<xrnet:MPReport xrnet:xsd="http://www.w3.org/2001/XMLSchema"
  xrnet:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xrnet:schemaVersion="1" xrnet:creationDate="2005-11-
18T09:20:56.0000000-05:00"
  xmlns:xrnet="iAnywhere.RfidNet.RFIDAnywhere">
<xrnet:report xrnet:reportName="MPReport"
  xrnet:reportID="91CA445D-C9F4-4CD3-85DC-9093EB403ED0"
  xrnet:date="2005-11-23T11:18:23.1328447-05:00"
  xrnet:totalMilliseconds="10004"
  xrnet:terminationCondition="DURATION">
<xrnet:reportDef>
  <xrnet:reportType>CURRENT</xrnet:reportType>
  <xrnet:DPPList>
    <xrnet:DPPitem xrnet:tagtype="RFIDTagEPC_Class0"
      xrnet:DPPid="RA_DPP" />
  </xrnet:DPPList>
</xrnet:reportDef>
<xrnet:groups>
  <xrnet:group name="Group1">
    <xrnet:groupDef>
      <xrnet:query>select * from sgtin96 where
CompanyPrefix=1</xrnet:query>
      <xrnet:groupCount></xrnet:groupCount>
      <xrnet:groupList>
        <xrnet:member id="303800004000008000000005"
          Uri="urn:epc:tag:sgtin-96:1.000001.0000002.5"
          Filter="1" CompanyPrefix="1" ItemReference="2"
          SerialNumber="5" Partition="SIX" Encoding="sgtin96" />
        <xrnet:member id="303800004000008000000006"
          Uri="urn:epc:tag:sgtin-96:1.000001.0000002.6"
          Filter="1" CompanyPrefix="1" ItemReference="2"
          SerialNumber="6" Partition="SIX" Encoding="sgtin96" />
        <xrnet:member id="303800004000020000000001"
          Uri="urn:epc:tag:sgtin-96:1.000001.0000008.1"
          Filter="1" CompanyPrefix="1" ItemReference="8"
          SerialNumber="1" Partition="SIX" Encoding="sgtin96" />
        <xrnet:member id="303800004000064000000005"
          Uri="urn:epc:tag:sgtin-96:1.000001.0000025.5"
          Filter="1" CompanyPrefix="1" ItemReference="25"
          SerialNumber="5" Partition="SIX" Encoding="sgtin96" />
        <xrnet:member id="303800004000078000000002"
          Uri="urn:epc:tag:sgtin-96:1.000001.0000030.2"
          Filter="1" CompanyPrefix="1" ItemReference="30"
          SerialNumber="2" Partition="SIX" Encoding="sgtin96" />
      </xrnet:groupList>
    </xrnet:groupDef>
  </xrnet:group>
</xrnet:groups>
</xrnet:report>
</xrnet:MPReport>

```

## Lesson 4: Stopping the connectors

The connectors must be stopped to end the simulation. They can be stopped from the Administrator Console. Recall that a report is generated and written to the file system every 10 seconds, so be sure to stop the connectors when they are not in use.

### ❖ To stop the simulation

1. Select the checkbox beside MPRServiceModule, MPRReport, FileOutput, and InvTracSim1, and then click Stop.

A red X appears beside the name of each connector once they have stopped.


## Properties of RFID MultiProtocol Report Definition connectors

The specific properties for defining a multiprotocol report, including the reader connectors to get raw data from, the type of tags to include, the report duration, and the RFID Anywhere messaging transport that will deliver the report are configured declaratively from the RFID Anywhere Administrator Console. The following properties are configured when defining a multiprotocol report connector from the RFID Anywhere Administrator Console:

Property	Description and Value
Collection Type	<p>Defines the type of tags to be included in the report.</p> <ul style="list-style-type: none"> <li>◆ <b>ADDITIONS</b> indicates that only tags that became observed during the report's duration are included in the report. This is the default value.</li> <li>◆ <b>DELETIONS</b> indicates that only tags that disappeared during the report's duration are included in the report.</li> <li>◆ <b>CURRENT</b> indicates that all tags that are observed during the reports duration are included in the report.</li> </ul> <p>This value is contained in the generated XML report in the &lt;xrnet:reportType&gt; element.</p>
Creation Date	<p>Indicates the date and time the report was created. This property value is generated automatically and cannot be edited.</p>
Description	<p>Provides a short description of the report that appears in the list of Local Services.</p>

Property	Description and Value
DPP	<p>Specifies the Data Protocol Processor (DPP) that is to be used to encode/decode different tag protocols in the format <i>&lt;Tag Protocol&gt;:&lt;DPP_to_use&gt;</i></p> <p>If RFIDTagEPC_Class0:RA_DPP is entered, the tags of protocol RFIDTagEPC_Class0 will be encoded and decoded using the RA_DPP Data Protocol Processor. This value is case sensitive.</p> <p>The default value for this property is RFIDTagEPC_Class0:RA_DPP.</p>
Duration	<p>Defines the length of the reporting cycle. Reports are generated based on tag activity during this period. Reports are generated continuously while the report connector running.</p> <p>Any valid duration with proper time units can be entered. Time units include:</p> <ul style="list-style-type: none"> <li>◆ <b>ms</b> milliseconds</li> <li>◆ <b>s</b> seconds</li> <li>◆ <b>m</b> minutes</li> <li>◆ <b>h</b> hours</li> <li>◆ <b>d</b> days</li> </ul> <p>For example, entering a value of 30s will generate a report every 30 seconds when the report engine is running. See additional report properties for configuring start and stop methods and conditions.</p> <p>This property is only used if the Start/Stop by Trigger property is set to False.</p>

Property	Description and Value
XML report encoding	<p>Specifies the encoding to use for the report output.</p> <p>The default value is UTF8 to indicate UTF-8 encoding. UTF16 specifies UTF-16 encoding.</p>
Return Table Layout	<p>If set to True, this property ensures that the underlying table definition of the tag data is included in the generated XML report within a &lt;xrnet:dataLayout&gt; element. If set to False (the default), the table layout is not included in the generated XML report.</p>
Include SQL in Report	<p>SQL statements are used to configure tag filtering and grouping (see Queries property). Setting this property to True includes the SQL statements in the generated XML report within an &lt;xrnet:query&gt; element. If set to False (the default), the SQL statements are not included in the XML report.</p>
Logical readers	<p>Identifies the logical readers that are the source of the raw RFID tag data for the report. Logical readers are defined within the Logical readers property of the MPRServiceModule component which manages raw data for all reports.</p> <p>For example, if the MPRServiceModule has a Logical readers property with a value Demo:MyReader (which signifies that the hardware connector MyReader is part of the logical reader with the arbitrary name Demo), the Logical reader property of this report definition would have a value of Demo.</p> <p>If multiple logical readers are to be used, enter each value on a new line in the Logical readers property field.</p>

Property	Description and Value
Name	<p>A unique name for the report, that appears in the list of Local Services. This value is also included in the generated XML report in the <code>xrnet:reportName</code> attribute of the <code>&lt;xrnet:report&gt;</code> element.</p>
Queries	<p>Used to filter and group the tag data to be included in the report. Each SQL query in this property forms a Group in the generated XML report.</p> <p>To create multiple groups in one report, enter multiple SQL queries in this property, separated by a carriage return.</p> <p>Specifying the Return Table Layout property as True during development enables the developer to see the table names (which map to specific encodings) and column names (which map to the partitions of the particular encoding).</p> <p>For example, the query <code>SELECT * FROM sgtin96 WHERE CompanyPrefix = 1</code> would ensure that only sgtin96-encoded tags generated by the company with the company prefix of 1 get included in the generated XML report.</p> <p> For more information on table and column names for the default DPP see <a href="#">“Using the default RFID Anywhere DPP for EPC tags”</a> [<i>RFID Anywhere Developer’s Guide</i>, page 88].</p>

Property	Description and Value
Report if empty	<p>Indicates whether or not to send empty reports. If set to True, empty reports are sent. If set to False (the default), empty reports are not sent.</p> <p>Depending on the logic of the application consuming and processing the generated XML reports, empty reports may have significant value.</p>
Version	<p>Indicates the version of the underlying report connector. This property can not be modified.</p>
Start Mode	<p>Indicates how the report connector should start.</p> <ul style="list-style-type: none"> <li>◆ <b>Manual</b> indicates that the connector will be started manually using a trigger, using the Start Time property, or by starting the connector from the RFID Anywhere Administrator Console. This is the default value.</li> <li>◆ <b>Automatic</b> indicates that the connector should start automatically when RFID Anywhere starts.</li> </ul>

Property	Description and Value
Start Operation	<p>If the Start mode property of the connector has been set to Automatic, this property indicates whether the connector's Start method or Configure method should be invoked.</p> <ul style="list-style-type: none"> <li>◆ <b>Run</b> Invokes the connector's Start method when RFID Anywhere is started, starting the connector and enabling it for use. This is the default value.</li> <li>◆ <b>Configure</b> Invokes the connector's Start method when RFID Anywhere is started, configuring the connector as specified in the method.</li> </ul>
Start Time	<p>Indicates when the connector should start creating reports. The value is entered in the format <i>Day:Hour:Minute</i> and allows the simple scheduling of reports.</p> <p>Supported values for <i>Day</i> include EveryDay, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, Weekday, and Weekend.</p> <p>Time units for <i>Hour</i> and <i>Minute</i> are based on the 24 hour clock.</p> <p>For example, configuring this property with a value of <i>Weekday:17:30</i> would start generating reports every weekday at 5:30 P.M. until a stop condition, such as the time specified in the Stop Time property, is reached.</p> <p>This property is only used if the Start/Stop by Trigger property is set to False.</p>
Status	<p>This property is used internally by RFID Anywhere and cannot be modified.</p>

Property	Description and Value
Stop Time	<p>Indicates when the connector should stop creating reports. The value is entered in the format <i>Day:Hour:Minute</i> and allows the simple scheduling of reports.</p> <p>Supported values for <i>Day</i> include EveryDay, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, Weekday, and Weekend.</p> <p>Time units for <i>Hour</i> and <i>Minute</i> are based on the 24 hour clock.</p> <p>For example, configuring this property with a value of <i>Weekday:7:30</i> would stop generating reports every weekday at 7:30 A.M.</p> <p>This property is only used if the Start/Stop by Trigger property is set to False.</p>
Subscribers	<p>Lists the transport connectors that generated XML reports should be sent to for delivery to enterprise systems. Multiple transport connectors should be separated by carriage returns.</p>
Trace Level	<p>Specifies the logging level for the connector. Supported values include Off, Error, Warning, Info and Verbose. The default value for this property is Verbose.</p>
Start/Stop by Trigger	<p>Indicates the ability of the connector to be started and stopped by using programmatic triggers.</p> <p>A value of True indicates that only triggers are used to start and stop a report cycle. In this case, the connector properties Duration, Start Time and Stop Time are not used.</p> <p>The default value for this property is False.</p>

## Summary

In this tutorial you learned how to use the Report Engine MP to generate processed information in the form of XML reports. Although Report Engine MP performs similar functions to the ALE standard, they are used for different purposes. Report Engine MP is used when processed reports are to be generated on ISO or EPC tags.

It is important to recognize that both Report Engine MP and ALE filter and group raw RFID data, but do not directly add business context. It is up to the external application that receives the reports to add business value and context.

The next chapter introduces you to another very important and useful method of applying business logic: custom business module development. A custom business module can be used to process raw RFID tags from readers and apply business logic. The major difference in this method is that the business logic is applied in the business module before any data is sent to an enterprise application. As a result, the enterprise application can be any dummy application that is simply capable of receiving the data from the business module.

Using components of the Inventory Tracker demo, the next chapter will provide examples of applying business logic to ALE reports and an example of using web services to access an RFID Anywhere business module method.



---

## CHAPTER 12

# Inventory Tracker Demo

### About this chapter

This chapter introduces the Inventory Tracker demo provided with RFID Anywhere. The Inventory Tracker demo demonstrates all three RFID Anywhere development models: Application Level Events (ALE), Report Engine MP, and custom business modules. The package includes external applications that provide examples of how to apply business logic using ALE reports and multiprotocol reports, how to view data sent from a custom business module, and how to use web services to access business module methods.

To this point, you have worked with Application Level Events (ALE) and Report Engine MP. This tutorial introduces you to custom business modules, which is another important development model. Custom business modules are edge processing components written with the .NET platform and integrated with RFID Anywhere. They are written to perform complex business logic at the edge. Unlike ALE or Report Engine MP, business modules can apply business logic and add business value prior to sending data to an enterprise application.

☞ To learn more about custom business module development and how to create your own business modules, see [“Using the RFID Anywhere Visual Studio Extension to Create Custom Business Modules”](#) [*RFID Anywhere Developer’s Guide*, page 7].

This chapter is divided into a number of distinct lessons:

- ◆ Introduction to the Inventory Tracker demo
- ◆ Setting up simulated connectors
- ◆ The ALE demo
- ◆ The Report Engine MP demo
- ◆ The business module demo
- ◆ Using web services to access business module methods
- ◆ Customizing the Inventory Tracker demo

---

<b>Topic:</b>	<b>page</b>
Introduction to the Inventory Tracker demo	205
Lesson 1: Setting up the simulator connectors	209
Lesson 2: The ALE demo	211
Lesson 3: The Report Engine MP demo	216
Lesson 4: The business module demo	221
Lesson 5: Using web services to access business module methods	226
Lesson 6: Customizing the Inventory Tracker demo	228
Installing the pre-configured Inventory Tracker demo	232
Summary	237

## Introduction to the Inventory Tracker demo

RFID Anywhere includes a complete demonstration package called Inventory Tracker. The package includes source code and simulated data for setting up and understanding the demonstrations. This lesson gives a high level overview of the package.

The entire package is stored in the *Inventory Tracker* directory of your RFID Anywhere installation.

☞ To learn how to customize components of the demo, see [“Lesson 6: Customizing the Inventory Tracker demo” on page 228](#).

### The simulation files

The package includes two RFID Simulator Data Editor files that define a basic series of tag appearances. Each file corresponds to the output of a reader at a fictitious location.

An XML file is also included that maps tag ID numbers to human-readable information. For example, if the simulator indicated that a tag with the ID 1.2.3.4 had appeared, the XML file could be used to determine that 1.2.3.4 is associated with a Class Two Widget manufactured by Widget Makers Inc. In a real-world scenario, this XML file might be replaced with a database or web service.

The custom business module demo, the ALE demo, and the Report Engine MP demo use the same simulated data and XML secondary source. Each demo processes the data differently to highlight the three development models of RFID Anywhere. This also showcases RFID Anywhere’s ability to apply a variety of different processes on RFID tags received from a single source. This is important because if you have an RFID reader that is hard-coded to process information for a particular enterprise application, you will need one reader for every application in your company. However, with RFID Anywhere, you can save money by using one RFID reader to communicate with a wide variety of enterprise applications.

### The ALE demo

The ALE demo showcases RFID Anywhere’s ALE development model. It generates filtered ALE reports and sends them to an external application. The external application then performs business logic to map raw tag IDs in the reports to actual product names and displays the result.

The ALE demo uses the RFID ALE Protocol business module and the ALE Event Cycle connector to generate a series of event cycle reports based on

the simulated data. The reports are sent via a TCP connector to the included ALE-driven .NET application.

The event cycle reports provide a variety of information to the application, including which tags have appeared or disappeared in addition to aggregate storage data. The ALE-driven application uses these reports in combination with the secondary source XML file to produce a visual representation of tag activity at the locations being monitored.

☞ For information on Application Level Events (ALE) and RFID Anywhere's ALE implementation, see [“Using ALE to Filter and View RFID Data” on page 153](#).

## The Report Engine MP demo

### Note

If you have followed and understood [“Using Report Engine MP” on page 183](#), then you can skip this demo. The steps in that chapter outline how a Report Engine MP connector can be configured to generate reports based on the simulated data of the Inventory Tracker Demo.

The Report Engine MP demo showcases RFID Anywhere's Report Engine MP development model. It uses simulated data to produce XML based reports on RFID tags. The demo shows how to send the reports as text files to the file system using the File messaging connector.

The Report Engine MP demo uses the RFID Multiprotocol Service Module and the RFID MultiProtocol Report Definition connector to generate a series of reports based on the simulated data.

☞ For information on Report Engine MP, see [“Using Report Engine MP” on page 183](#).

## The custom business module demo

The custom business module demonstration showcases RFID Anywhere's third development model: custom business modules. It uses a custom business module called InventoryTracker to process simulated data and add business context information from the secondary source XML file. It then provides the results to two enterprise applications in the form of custom XML reports.

The first enterprise application is a .NET program that receives the XML reports from the InventoryTracker via a TCP connector. It processes the reports and maintains a display of which assets have been detected and at which locations.

The custom business module demo also exposes a method via web services for retrieving a current inventory list.

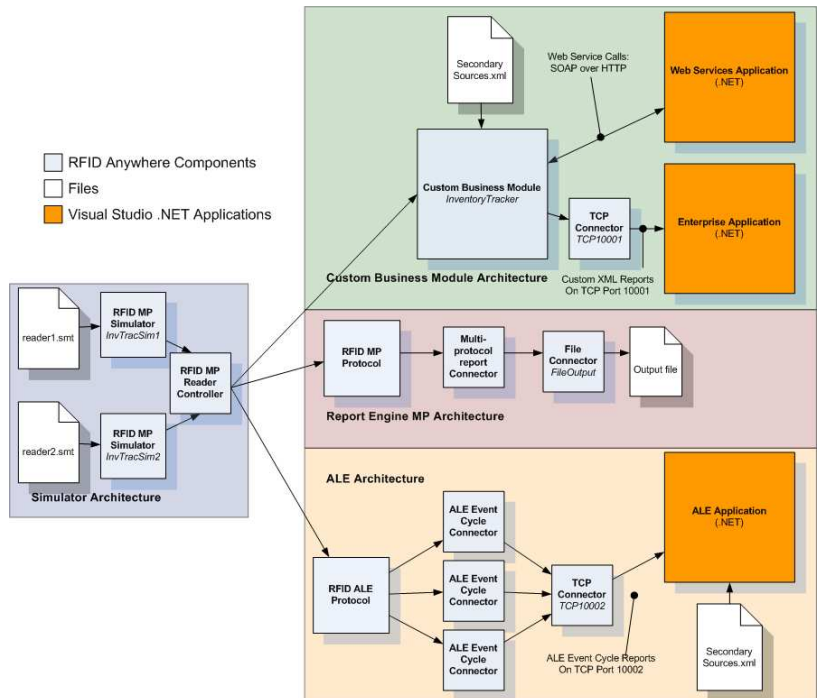
## Using web services to access business module methods

The Inventory Tracker business module exposes a method to the web called `GetCurrentInventory()`. To access this method from outside of the RFID Anywhere environment, you need to access the automatically-generated WSDL file and include all of the necessary security parameters. The web services demo uses an external application to access `GetCurrentInventory()`.

The application contains fields for entering the user credentials, which are inserted into the SOAP request in order to establish communication with the RFID Anywhere service. The results received from the Inventory Tracker business module are displayed in a text box.

☞ For information on exposing web methods in your business module, see [“Obtaining complete EPC tag URI from the default DPP” \[RFID Anywhere Developer’s Guide, page 111\]](#).

The following diagram illustrates how the various components of the Inventory Tracker demo interact with one another.



The next several lessons provide detailed instructions on configuring and viewing the Inventory Tracker demos. They introduce you to various components of RFID Anywhere and provide you with a much deeper understanding of all three RFID Anywhere development models. It is therefore highly recommended that you follow these lessons.

**Note**

If after following the lessons you are unable to get the demos working, or if you want to view the demos first before reading the lessons, see [“Installing the pre-configured Inventory Tracker demo” on page 232](#) to install the Inventory Tracker demo package with pre-configured connectors.

## Lesson 1: Setting up the simulator connectors

This lesson explains how to add and configure the required RFID Multiprotocol (MP) Simulator connectors.

### Starting RFID Anywhere

The RFID Anywhere service must be started before you can begin.

#### ❖ To start the RFID Anywhere service

1. From the Start menu, choose Programs ► RFID Anywhere ► Start Service.

The RFID Anywhere service starts.

2. At the Administrator Console Login page, type your credentials and then click Log On.

The Home page opens.

### Creating the RFID Multiprotocol Simulator connectors

The RFID Anywhere demo package uses two RFID MP Simulator connectors to read files containing simulated RFID data.

#### Note

If you have completed the tutorial “[Lesson 3: Configuring the RFID Multiprotocol Simulator connector](#)” on page 150 and have already created the InvTracSim1 RFID MP Simulator connector, then skip to step 10 in the Configuring the MP simulator connectors section below.

#### ❖ To create the RFID Multiprotocol Simulator connectors

1. In the Administrator Console Service Type dropdown list, choose RFID Multiprotocol Simulator, and then click Add.

An RFID Multiprotocol Simulator connector appears in the Local Services list.

2. Repeat step 1 to create another RFID Multiprotocol Simulator connector.

The second RFID Multiprotocol Simulator connector appears in the Local Services list.

### Configuring the Multiprotocol simulator connectors

Now that you have created two RFID Multiprotocol Simulator connectors,

you must configure the connectors with the appropriate smoothing settings and provide simulated data files.

❖ **To configure the simulator connectors**

1. In the Administrator Console Local Services list, select one of the two RFID MP Simulators you created, and then click Edit.

The Properties Manager for the RFID MP Simulator appears to the right.

2. Expand the Antenna1 ► Smoothing set of properties.
3. In the textbox for the Antenna1 ► Smoothing ► Glimpsed Timeout property, type **100ms** (milliseconds).
4. In the textbox for the Antenna1 ► Smoothing ► Lost Timeout property, type **100ms** (milliseconds).
5. In the textbox for the Antenna1 ► Smoothing ► Observed Threshold property, type **100ms** (milliseconds).
6. In the textbox for the Antenna1 ► Smoothing ► Observed Timeout property, type **100ms** (milliseconds).
7. In the textbox for the Cycles Per Trigger property, type **1**.
8. In the dropdown list for the Read on Start property, choose True.
9. In the textbox for the Name property, type **InvTracSim1**.
10. Beside the textbox for the Session Data property, click Browse and locate the file *reader1manual.smt* located in the *Inventory Tracker\SimulatedData* directory of your RFID Anywhere install.  
If you did not create *reader1manual.smt* in “[Creating Simulated RFID Data](#)” on page 137, then use *reader1.smt* located in the *Inventory Tracker\SimulatedData* directory of your RFID Anywhere install.
11. Click the green disk in the Properties Manager title bar to save the above changes.  
The InvTracSim1 Multiprotocol Simulator is updated.
12. Repeat steps 1 through 9 for the other RFID Multiprotocol Simulator using the same settings, but this time naming it **InvTracSim2** and using **SimulatedData\reader2.smt** for the Session Data.

## Summary

You have configured RFID Anywhere to provide simulated data through two RFID MP Simulator connectors. You are now ready to set up the custom business module and ALE demos.

## Lesson 2: The ALE demo

The ALE demo uses the RFID ALE Protocol to generate a series of event cycle reports based on the simulated data. It uses three ALE Event Cycle connectors to generate the processed information (reports), which are then sent via a TCP connector to the included ALE-driven .NET application.

This lesson explains how to set up the ALE demo.

### Configuring the RFID ALE Protocol business module

You need to configure the RFID ALE Protocol business module to use the ALE demo.

#### ❖ To configure the RFID ALE Protocol business module

1. From the Administrator Console Home page, select the ALEServiceModule in the Local Services list and then click Edit.  
The Properties Manager for the RFID ALE Protocol appears to the right.
2. Expand the Logical Readers property. In the text field, type **Source1:InvTracSim1** and press the green plus sign to add the entry to the list. In the new text field, type **Source2:InvTracSim2** and press the green plus sign to add the entry to the list.
3. Click the green disk in the Properties Manager title bar to save the above changes.

The RFID ALE Protocol business module is updated.

#### Note

It is also possible to create a combined source in the RFID ALE Protocol that creates a logical name for more than one physical source. For example, you could add both **CombinedSource:InvTracSim1** and **CombinedSource:InvTracSim2** to the Logical Readers property.

In ALE Event Cycle Report definitions, using the name CombinedSource would refer to both InvTracSim1 and InvTracSim2. This simplifies the process of adding new physical data sources by not requiring ALE Event Cycle Report definitions to be changed.

### Creating and configuring a Tcp messaging connector

The ALE demo requires a Tcp messaging connector to communicate with the enterprise application.

**Note**

If you have completed the tutorial [“Adding the Tcp messaging connector” on page 161](#) and have already created the InvTracAleTCP Tcp messaging connector, then you can skip this section.

❖ **To create and configure the Tcp messaging connector**

1. Choose Tcp Messaging from the Service Type dropdown list. Click Add.  
A Tcp messaging connector is added to the Local Services list.
2. Select the Tcp messaging connector in the Local Services list, and then click Edit.  
The Properties Manager for the Tcp messaging connector appears to the right.
3. In the textbox for the IP Address property, type **localhost**.
4. In the textbox for the Name property, type **InvTracAleTCP**.
5. In the textbox for the Port property, type **10002**.
6. Click the green disk in the Properties Manager title bar to save the above changes.  
The Tcp messaging connector is updated.

## Creating and configuring ALE Event Cycle connectors

The demo uses three ALE Event Cycle connectors to define the reports that are sent to the ALEDemo enterprise application.

❖ **To create and configure the ALE Event Cycle connectors**

1. Choose ALE Event Cycle from the Service Type dropdown list and then click Add.  
A new ALE Event Cycle connector appears in the Local Services list.
2. Select the new ALE Event Cycle connector and then click Edit.  
The Properties Manager for the ALE Event Cycle connector appears to the right.
3. Expand the Event Cycle set of properties. Expand the Start stop conditions set of properties. In the textbox for the Duration (ms) property, type **5000**.  
Reports will contain tag information on those tags that were inserted or removed from the reader’s range in the past 5 seconds.

4. Expand the Logical Readers property. In the text field, type **Source1** and press the green plus sign to add the entry to the list.

The ALE Event Cycle connector will report on data received from RFID readers that are under the Source1 group.

5. In the textbox for the Report(s) Definition property, type **2**.

The number of report definitions expands to two with the creation of the reportSpecs(1) and reportSpecs(2) sets of properties.

6. Expand the reportSpecs(1) set of properties. In the textbox for the reportSpecs(1) ► Report Name property, type **Source1Additions**.

7. In the dropdown list for the reportSpecs(1) ► EPC to Be Reported property, choose ADDITIONS.

This report definition will only contain tag information of those tags that were inserted in the past 5 seconds.

8. Expand the reportSpecs(2) set of properties. In the textbox for the reportSpecs(1) ► Report Name property, type **Source1Deletions**.

9. In the dropdown list for the reportSpecs(2) ► EPC to Be Reported property, choose DELETIONS.

This report definition will only contain tag information on those tags that were removed in the past 5 seconds.

10. In the textbox for the Name property, type **InvTracSource1ALE**.

11. In the dropdown list for the Send Reports as Object property, choose True.

12. Expand the Subscribers property. In the text field, type **InvTracAleTCP** and press the green plus sign to add the entry to the list.

The processed ALE reports will be sent to external applications via the InvTracAleTCP.

13. Click the green disk in the Properties Manager title bar to save the above changes.

The ALE Event Cycle connector is updated.

To create the second ALE Event Cycle, repeat steps 1 through 12 using the following properties:

- ◆ **Duration (ms)** 5000
- ◆ **Logical Readers** Source2

- ◆ **Report Definitions** 2
  - **First Report** Enter the following values:
    - ◆ **Report Name** Source2Additions
    - ◆ **EPC to Be Reported** ADDITIONS
  - **Second Report** Enter the following values:
    - ◆ **Report Name** Source2Deletions
    - ◆ **EPC to Be Reported** DELETIONS
- ◆ **Name** InvTracSource2ALE
- ◆ **Send Reports as Object** True
- ◆ **Subscribers** InvTracAleTCP

To create the third ALE Event Cycle connector, repeat steps 1 through 10 using the following properties. Note that this connector uses only one report definition and includes two sources in its Logical Readers property.

- ◆ **Duration (ms)** 5000
- ◆ **Logical Readers** Source1  
Source2
- ◆ **Report Definitions** 1.  
Enter the following values:
  - **Grouping Patterns** urn:epc:pat:sgtin-96:X.X.X.\*
  - **Include Count** True
  - **Include Tag Uri** False
  - **Report If Empty** True
  - **Report Name** CountReport
  - **EPC to Be Reported** CURRENT
- ◆ **Name** InvTracCountALE
- ◆ **Send Reports as Object** True
- ◆ **Subscribers** InvTracAleTCP

**Note**

This demonstration uses more ALE reports and connectors than would normally be required. This is done to show a broad range of ALE features.

## Starting the ALE demonstration

You have now configured the components required to run the ALEDemo application.

**❖ To start the demonstration**

1. In the Administrator Console, select and restart the following components:
  - ◆ The three ALE Event Cycle connectors (InvTracSource1ALE, InvTracSource2ALE, and InvTracCountALE)
  - ◆ The InvTracAleTcp messaging connector
  - ◆ InvTracSim1 and InvTracSim2 MP Simulators
  - ◆ The ALEServiceModule
2. From your RFID Anywhere installation directory, open the *Inventory Tracker* folder and then double-click *AleDemo.exe*.

The AleDemo application appears with the list of assets pulled from the secondary source XML file. As ALE event cycle reports are received by the AleDemo application over TCP, the application processes the reports and updates the location of each asset.

3. Once the demo starts, you can stop the demo by selecting all of the components listed in step 1 and clicking Stop. Close the Ale Demo application.

**Note**

You can change AleDemo application settings by modifying the *config.ini* file. This file is located in the *Inventory Tracker\ALEApplication\bin\Release* folder of your RFID Anywhere installation directory.

☞ See “[Configuring the ALE application](#)” on page 231 for more information.

## Lesson 3: The Report Engine MP demo

### Note

If you have followed and understood the chapter [“Using Report Engine MP” on page 183](#), then you can skip this demo. The steps in that chapter outline how a Report Engine MP connector can be configured to generate reports based on the simulated data of the Inventory Tracker Demo.

This tutorial makes use of multiple RFID Anywhere connectors. These include a File messaging connector, a MultiProtocol Report Definition connector, and a RFID Multiprotocol Simulator connector.

### The RFID Multiprotocol Simulator connector

The RFID Multiprotocol Simulator connector simulates the appearance of the tags that were defined using the RFID Simulator Data Editor.

### Note

If you have not already created the InvTracSim1 RFID Multiprotocol Simulator connector, see [“Lesson 3: Configuring the RFID Multiprotocol Simulator connector” on page 150](#) for instructions on creating the connector.

### The File messaging connector

The File messaging connector allows processed information in the form of files to be saved to the file system. This connector also allows the processed information from RFID Anywhere to be easily viewed.

### Note

If you already created the File messaging connector named FileOutput from [“Adding the File messaging connector” on page 160](#), you can skip this section.

#### ❖ To add and configure the File messaging connector

1. In the Administrator Console Home page, choose File Messaging from the Service Type dropdown list and then click Add.

The File messaging connector is added to the list of Local Services and is highlighted. The new File messaging connector has the default name FileConnector1.

2. Edit the properties of the new connector by selecting the checkbox beside it and then clicking Edit.

The properties for the connector appear on the right side of the window.

3. In the textbox for the Name property, type **FileOutput**.
4. In the textbox for the Directory property, enter **C:\RFID Output** to specify the folder where the files from the simulation output is stored.  
This directory is created on your computer (if it does not already exist) when you start the connector.
5. Click the green disk in the Properties Manager title bar to save the above changes.

The File messaging connector is updated.

## The RFID MultiProtocol Report Definition connector

The RFID MultiProtocol Report Definition connector performs similar functions to the ALE Event Cycle connector described in [“Using ALE to Filter and View RFID Data” on page 153](#). The RFID MultiProtocol Report Definition connector contains specific properties for defining a multiprotocol report, including the reader connectors to get raw data from, the type of tags to include (filtering), the report duration and the RFID Anywhere messaging transport(s) that will deliver the reports. The connector will generate reports based on these property values.

As in the ALE tutorial, you will filter out the raw RFID data before sending the reports. For the purposes of this tutorial, you will filter the data such that only products from Robertson Health are included in the reports. There are eight products shipped from Robertson Health in the simulation data and their tag IDs all have a Company Prefix of 1.

### ❖ To add and configure a RFID MultiProtocol Report Definition connector

1. Choose MultiProtocol Report Definition from the Service Type dropdown list and then click Add.

The RFID MultiProtocol Report Definition connector is added to the list of Local Services and is highlighted. The new connector has the default name MPReport1.

2. Edit the properties of the new connector by selecting the checkbox beside it and then clicking Edit.

The properties for the connector appear on the right side of the window.

3. In the dropdown list for the Collection Type property, choose CURRENT.


4. Expand the DPP property. In the text field, type **RFIDTagEPC\_Class0:RA\_DPP** and press the green plus sign to add the entry to the list.

This value must be the same as the value used in the RFID Simulator Data Editor when creating the tags. This property is case sensitive.
5. In the text field for the Duration property, type **10s** to generate a report every 10 seconds.
6. In the dropdown list for the Return Table Layout property, choose False.

Setting this property to True ensures that the underlying table definition of the data is included in the generated XML report within a `<xrnet:dataLayout>` element. When set to False, the table layout is not included in the generated report.
7. In the dropdown list for the Include SQL in Report property, choose True.
8. Expand the Logical Readers property. In the text field, type **MPSource**.
9. In the textbox for the Name property, type **MPReport**.
10. Expand the Queries property. In the text field, type **SELECT \* FROM sgtin96 WHERE CompanyPrefix = 1** and press the green plus sign to add the entry to the list.

This property is used to filter and group the tag data to be included in the report. The value entered for this property are SQL queries. As mentioned earlier, you want to filter the data such that you only include products from Robertson Health in the report. All products shipped from Robertson Health have a Company Prefix of 1.

To create multiple groups in one report, enter multiple SQL queries in this property, each as a new item.

 For more information on the table and column names used by the default RFID Anywhere DPP, see [“Using the default RFID Anywhere DPP for EPC tags” \[RFID Anywhere Developer’s Guide, page 88\]](#).
11. Expand the Subscribers property. In the text field, type **FileOutput** and press the green plus sign to add the entry to the list.
12. Click the green disk in the Properties Manager title bar to save the above changes.

The RFID MultiProtocol Report Definition connector is updated.

## The RFID Multiprotocol Service Module

The RFID Multiprotocol Service Module provides the business logic to link the reader group definition to the reader that is gathering data (RFID Multiprotocol Simulator).

The RFID Multiprotocol Service Module is automatically installed in the list of Local Services and has the name MPRServiceModule.

### ❖ To configure the RFID Multiprotocol Service Module

1. Edit the properties of the RFID Multiprotocol Service Module by selecting the checkbox beside it and then clicking Edit.  
The properties for the connector appear on the right side of the window.
2. Expand the Logical Readers property. In the text field, type **MPSource:InvTracSim1** and press the green plus sign to add the entry to the list.
3. Click the green disk in the Properties Manager title bar to save the above changes.

The RFID Multiprotocol Service Module is updated

## Starting the Report Engine MP demo

You must start the connectors you have created to begin the simulation. Connectors are started from the Administrator Console.

### ❖ To start the demo

1. Select the checkbox beside MPRServiceModule, MPReport, FileOutput, and InvTracSim1, and then click Restart.  
The connectors are identified with green arrows once they start. If an error message appears above the list of Local Services, ensure that you have followed the steps correctly and have not made any spelling errors.

## Viewing the simulation reports

As stated earlier, the File messaging connector is used to send processed information (reports) in the form of files to a user-specified location in the file system. It can be configured to save the files in any format, such as the .txt format. When configured to send files in .txt format (such as in the File messaging connector named FileOutput), one advantage of the File messaging connector is that the files can be easily accessed and viewed via a generic text editor.

❖ **To view the processed information via a text editor**

1. In the Administrator Console, if they are not already started, select MPRServiceModule, MPReport, FileOutput and InvTracSim1, and then click Restart.

2. Browse to *C:\RFID Output*.

You configured the File Messaging Connector to save reports in this folder when you set the connector's properties. Since in the RFID MultiProtocol Report Definition connector, you specified the Duration property as 10s, a new text file is created in this folder every 10 seconds.

Each file contains the tags (if any) that were present during the previous second. The name of each text file is in the format *file-dd-mm-yyyy-hh-mm-ss-number.txt*.

3. Open the first file that appeared using a text editor.

The file includes the filtered tags that have appeared since the last cycle.

**Note**

Reports from Report Engine MP contain the various attributes of the tags based on the Queries property. Since the simulation tags are in the EPC SGTIN-96 format and the Queries property used \* to indicate that all columns should be included in the report, the reports list the Filter, Company Prefix, Product Name, and the Serial Number as separate attributes for each tag.

Note that the

## Stopping the connectors

The connectors must be stopped in order to end the simulation. They can be stopped from the Administrator Console. Recall that a report is generated and written to the file system every 10 seconds, so be sure to stop the connectors when they are not in use.

❖ **To stop the simulation**

1. Select the checkbox beside MPRServiceModule, MPReport, FileOutput, and InvTracSim1, and then click Stop.

A red X appears beside the name of each connector once they have stopped.

## Lesson 4: The business module demo

This lesson explains how to set up the custom business module demonstration.

### Deploying the Inventory Tracker custom business module

To load the Inventory Tracker business module to the Administrator Console Local services list, you must build it using Visual Studio .NET. When a business module is built, it is automatically deployed to the RFID Anywhere service running on the local computer.

#### Note

If you do not have Microsoft Visual Studio installed on your computer, then proceed to the section [“Using Provisioning to deploy the InventoryTracker business module”](#) on page 222.

To use the following procedure, you must have the RFID Anywhere Visual Studio Extension tool installed on your computer.

#### ❖ To deploy the InventoryTracker business module using Microsoft Visual Studio

1. In the *Inventory Tracker\InventoryTracker\SourceCode* directory of your RFID Anywhere installation, open the Visual Studio C# project file *InventoryTracker.csproj*.

The InventoryTracker project opens in Visual Studio.

If RFID Anywhere is installed to the default location (*C:\Program Files\Sybase\RFID Anywhere\*), then proceed to skip to step 7.

2. Choose Project ► Add Reference.
3. Click Browse.
4. Go to the *Bin* folder of your RFID Anywhere installation directory.  
The *Bin* folder contains all of the RFID Anywhere namespaces and connector DLL files.
5. Select the components that are missing and then click Open. The following list includes some of the components that may be missing if you installed RFID Anywhere to a non-default directory.
  - ◆ *iAnywhere.RfidNet.Rfid.dll*
  - ◆ *iAnywhere.RfidNet.Core.dll*
  - ◆ *iAnywhere.RfidNet.DPP.dll*
  - ◆ *iAnywhere.RfidNet.DPP.RA.dll\**

- ◆ *iAnywhere.RfidNet.Common.dll*
- ◆ *iAnywhere.RfidNet.Rfid.Tags.dll*
- ◆ *iAnywhere.RfidNet.Rfid.Multiprotocol.dll*
- ◆ *iAnywhere.RfidNet.OIM.dll*

\*This file is located in the *Bin\DPP* folder.

6. Click OK. Accept any alert messages that appear.
7. Build the project in Visual Studio by choosing Build ► Build Solution.  
The Inventory Tracker business module DLL is added to the *bms* folder of your RFID Anywhere installation directory.
8. The RFID Anywhere service must be restarted to register the InventoryTracker business module and load it in the Administrator Console.
  - a. Close the Administrator Console.
  - b. From the Start menu, choose Programs ► RFID Anywhere ► Stop Service.  
The RFID Anywhere service stops.
  - c. From the Start menu, choose Programs ► RFID Anywhere ► Start Service.  
The RFID Anywhere service starts.
  - d. Once the RFID Anywhere service is running, open the Administrator Console and log in.  
The InventoryTracker business module is loaded under the Home tab.

## Using Provisioning to deploy the InventoryTracker business module

- ❖ **To use provisioning to deploy the InventoryTracker business module**
  1. Choose Start ► Programs ► RFID Anywhere ► Administrator Console.  
The Administrator Console opens.
  2. At the Login page, type your Windows login name and password for the User Name and Password fields, respectively. Click Log On.  
The Home page opens.
  3. Click the Provisioning tab in the Administrator Console.
  4. Under Deploy New Packages, click Browse.

5. Browse to the *Inventory Tracker\InventoryTracker\BM\_MSI* directory located in your RFID Anywhere installation directory.
6. Select the *InventoryTrackerSetupBM.msi* file, and then click Open.
7. Click Deploy to deploy the business module.  
The business module is queued for installation.
8. Click Refresh on the Provisioning tab to update the status of the InventoryTracker business module.

**Note**

The RFID Anywhere service automatically restarts when a business module is deployed, so allow the service time to restart before clicking Refresh. An error message may appear if Refresh is clicked while the service is restarting.

9. Once the status is updated to Installed, click Home. If the InventoryTracker business module is not visible under Local Services, click Refresh.

## Creating and configuring the Tcp messaging connector

The InventoryTracker custom business module requires a Tcp messaging connector to communicate with the EnterpriseApp enterprise application.


### ❖ To create and configure the Tcp messaging connector

1. Open the Administrator Console by choosing Start ► Programs ► RFID Anywhere ► Administrator Console.  
The Administrator Console opens.
2. On the Login page, type your Windows login name and password for the User Name and Password fields, respectively. Click the Log On.  
The Home page opens.
3. In the Administrator Console Service Type dropdown list, choose Tcp Messaging, and then click Add.  
A Tcp messaging connector is added to the Local Services list.
4. Select the Tcp messaging connector in the Local Services list, and then click Edit.  
The Properties Manager for the Tcp messaging connector appears to the right.
5. In the textbox for the IP Address property, enter **localhost**.

6. In the textbox for the Name property, type **InvTracBMTCP**.
7. In the textbox for the Port property, type **10001**.
8. Click the green disk in the Properties Manager title bar to save the above changes.  
The Tcp messaging connector is updated.

## Configuring the InventoryTracker business module

### ❖ To configure the InventoryTracker business module

1. In the Local Services list, select the InventoryTracker custom business module, and then click Edit.  
The Properties Manager for the InventoryTracker business module appears to the right.
2. Expand the Logical Readers property. In the text field, type **InvTracSim1** and press the green plus sign to add the entry to the list. In the new text field, type **InvTracSim2** and press the green plus sign to add the entry to the list.
3. In the textbox for the SecondarySourceXML property, type **Default**, or enter the full path of a secondary source file. This file is used in place of a database and contains the mapping of raw tag data to actual manufacturer and product names for display.  
The Default value configures the business module to use the default *SecondarySources\SecondarySource.xml* file located in the *Inventory Tracker* directory of your RFID Anywhere installation.  
Click the green disk.  
 For information on modifying the sample data used by the business module see, [“Changing the simulated data” on page 228](#).
4. Expand the Subscribers property. In the text field, type **InvTracBMTCP** and press the green plus sign to add the entry to the list.
5. Click the green disk in the Properties Manager title bar to save the above changes.  
The InventoryTracker business module is updated.

## Starting the business module demonstration

You have now set up the necessary components to run the InventoryTracker custom business module demonstration.

**❖ To start the demonstration**

1. In the Local Services list of the Administrator Console, select the following local services and click Restart:

- ◆ InventoryTracker
- ◆ InvTracBMTCP
- ◆ InvTracSim1
- ◆ InvTracSim2

2. From your RFID Anywhere installation directory, open the *Inventory Tracker* folder and then double-click *EnterpriseApp.exe*.

The EnterpriseApp application appears with lists of tags appearing and disappearing at each simulated reader. The business module sends periodic updates to this application for display. The business module mapped raw tag data to specific company and product names according to the secondary source XML file, so the EnterpriseApp is simple responsible for receiving the business module output over TCP and displaying it appropriately.

3. Once the demo is starts, you can stop the demo by selecting all of the components listed in step 1 and clicking Stop. Close the Enterprise application.

**Note**

You can change the EnterpriseApp application settings by modifying the *config.ini* file. This file is located in the *Inventory Tracker\EnterpriseApp\bin\Release* folder of your RFID Anywhere installation directory.

☞ See “[Configuring the Enterprise application](#)” on page 230 for more information.

## Lesson 5: Using web services to access business module methods

Web services is a software system designed for interoperability between applications running on different computers. RFID Anywhere allows business modules to expose methods via web services. External applications running on different networks can access these web methods to retrieve rich RFID data.

The web services demo provides an example of how an external application can be used to access a web method. The application accesses the `GetCurrentInventory()` method exposed by the InventoryTracker business module via web services. `GetCurrentInventory()` method gets the list of tags that are visible by all readers listed under the Source property. The method returns a string that contains the serial number, product name (SKU), company name, and reader name for each tag.

### Starting the web services demonstration

You will use the Inventory Tracker business module and both RFID MP Simulator connectors in this demo. The business module demo must be properly configured and started before proceeding.

**Note**

If you have not already configured the Business module demo, see [“Lesson 3: Configuring the RFID Multiprotocol Simulator connector”](#) on page 150 for instructions.

❖ **To view the demonstration**

1. In the Administrator Console Home page, select the Inventory Tracker business module, and both the InvTracSim1 and InvTracSim2 RFID MP Simulator connectors from the Local Services list. Click Restart.
2. You will need to enter valid user credentials to use the Web services application. The user must be a member of RfidWebServiceUsers. To give RfidWebServiceUsers permission to an user account:
  - a. Log in to the RFID Anywhere Administrator console as a Windows Administrator
  - b. Click the Local Security tab
  - c. Under the Users list, select the pencil beside the user account you want to modify
  - d. Select the checkbox in the RfidWebServiceUsers column

- e. Click the green disk to save changes
3. From your RFID Anywhere installation directory, open the *Inventory Tracker* folder and then double-click *WebServicesApp.exe*.  
The web services application contains fields for entering user credentials. This includes domain, username, and password. The application will enter this information in the SOAP request, and the RFID Anywhere service (the recipient) will verify its authenticity.
4. Enter your username, password, and domain. Note that the password and domain fields are case sensitive.
5. Click Get Tags.  
You can see the received tag information in the Results text box. For each tag that is currently in front of a reader, the data contains the serial numbers, product names, company names and the reader names, respectively.

❖ **To stop the demonstration**

1. Select the Inventory Tracker business module, and both the InvTracSim1 and InvTracSim2 RFID MultiProtocol Simulator connectors from the Local Services list. Click Stop.
2. Close the web services application.

☞ For information on exposing methods from your business modules via web services, see [“Obtaining complete EPC tag URI from the default DPP” \[RFID Anywhere Developer’s Guide, page 111\]](#).

## Lesson 6: Customizing the Inventory Tracker demo

**Note**

This lesson is optional.

The RFID Anywhere demo package was designed with flexibility in mind. It includes a number of *.ini* files, XML documents, and OIM properties for configuring and customizing the demos. This document explains how to use these resources.

### Changing the simulated data

The SimulatedData folder contains two RFID Anywhere Simulator Data Editor files corresponding to the two storage locations: *reader1manual.smt* (or *reader1.smt*) and *reader2.smt*. So long as the EPC SGTIN-96 encoding is used, the simulated data in these files can be modified. Other encodings, especially 64-bit, are not recommended since all of the demo code assumes the SGTIN-96 encoding is used.

Whenever new tags are added to the simulated data, the same tags must be added to the XML document found in the *SecondarySources* folder. For example if the tag 1.1.1.1 was added to the simulated data, it might be inserted into the XML document. The product names and manufacturers associated with existing tags in the demo can also be modified through this XML document to easily demonstrate the tracking of assets for other industries if required for demonstration purposes.

A section of the XML file is below:

```
<?xml version="1.0" encoding="us-ascii" standalone="yes"?>
<AssetCollection>
  <Asset>
    <ID>1.1.2.5</ID>
    <Company>Robertson Health</Company>
    <SKU>AutoClean 77 Sterilizer</SKU>
  </Asset>
  <Asset>
    <ID>1.1.2.6</ID>
    <Company>Robertson Health</Company>
    <SKU>AutoClean 77 Sterilizer</SKU>
  </Asset>
  <Asset>
    <ID>1.1.2.7</ID>
    <Company>Robertson Health</Company>
    <SKU>AutoClean 77 Sterilizer</SKU>
  </Asset>
</AssetCollection>
```

## Adding real hardware to Inventory Tracker

Although the Inventory Tracker demo used two simulation files to feed data into the various components of the demo, adding real RFID readers and tags is simple. Migrating from simulation files to actual RFID hardware drives home the flexibility of the RFID Anywhere architecture and demonstrates the common development process of building and testing an application using simulated data and then moving to actual RFID hardware for deployment.

This section outlines the steps for configuring an RFID reader and connector to replace one of the RFID Multiprotocol Simulator connectors. In this demo, the actual RFID reader tracks inventory in the first location, while the remaining RFID Multiprotocol Simulator tracks inventory in the second location.

The following steps are required to update the Inventory Tracker demo to use actual RFID hardware:

1. Create RFID tags for use with the RFID hardware that will be used. To eliminate the need to modify components of the demo to use a different DPP or different data, encode the tags as EPC SGTIN-96 and use the same raw data that was used when creating the simulation data in [“Creating Simulated RFID Data” on page 137](#).
2. Configure the appropriate hardware connector for the hardware that will be used. Smoothing thresholds of 2 seconds are appropriate for this demo. Give the connector a unique name. For this section, it is assumed the name of the connector attached to the actual RFID hardware is **InvTracReader**.
3. To have the Inventory Tracker business module receive data from the InvTracReader connector instead of the InvTracSim1 connector, open the Properties Manager for the Inventory Tracker business module and edit the Logical Readers property. Replace the entry `InvTracSim1` with `InvTracReader`. Save the changes to the business module.
4. To have the ALE Event Cycles receive data from the InvTracReader connector instead of the InvTracSim1 connector, open the Properties Manager for the ALEServiceModule business module and edit the Logical Readers property. Replace the entry `Source1:InvTracSim1` with `Source1:InvTracReader`. Save the changes to the ALEServiceModule.

Recall that the separate ALE event cycle connectors were configured to receive data from the logical reader named Source1. By updating the

Logical Readers property of the higher-level ALEServiceModule to indicate that Source1 now includes InvTracReader instead of InvTracSim1, no changes need to be made to the individual event cycle connectors.

5. To have the Report Engine MP report data received from the InvTracReader connector instead of the InvTracSim1 connector, open the Properties Manager for the RFID Multiprotocol Service Module business module and edit the Logical Readers property. Replace the entry `MPSource:InvTracSim1` with `MPSource:InvTracReader`. Save the changes to MPRServiceModule.

Recall that the Report Engine MP report definition connector was configured to receive data from the logical reader named MPSource. By updating the Logical Readers property of the higher-level RFID Multiprotocol Service Module to indicate that SourceMP now includes InvTracReader instead of InvTracSim1, no changes need to be made to the individual report definition connector or any other report definition connectors you may have configured to receive data from MPSource.

6. Unless the data encoded on the tag is not EPC SGTIN-96 with the values identified in the `SecondarSourceXML` file, no changes need to be made to the Enterprise Application or ALE Application.

Restart all of the connectors and business modules, including the hardware connector, to start using actual RFID hardware to monitor location 1, and using simulation to monitor location 2.

## Configuring the Enterprise application

The Enterprise Application includes a file named `config.ini` in the same directory as its executable—namely in the `bin/release` folder of the EnterpriseApp Visual Studio project. By editing the `config.ini` file with a text editor, a number of properties can be modified:

- ◆ **logo** specifies the GIF logo to display in the application. The application looks for the image in the same directory as the executable itself. By default, the image is a custom iAnywhere logo with the file name `corporateLogo.gif`.
- ◆ **left and right** specify the arbitrary names of the locations where the simulated RFID readers have been placed. These names appear on the application itself.
- ◆ **port** specifies the port on which the application listens for incoming XML reports from the InventoryTracker custom business module. The default port for this application is 10001.

**Note**

If you want to change location or file name of the XML secondary source document for the Enterprise Application, you must set the InventoryTracker custom business module's SecondarySourceXML OIM property.

## Configuring the ALE application

The ALE application uses a similar *config.ini* file. In addition to the logo, left, right, and port properties, it also contains two additional properties:

- ◆ **unknown** defines what the application displays if an asset is not found at either of the simulated locations.
- ◆ **secondarySource** specifies the location of the secondary source XML document to be used by the application for mapping raw tag data to specific product and company names for display.

## Installing the pre-configured Inventory Tracker demo

The previous lessons in this chapter introduced the Inventory Tracker package, and provided instructions for configuring and viewing the different demonstrations. If after following the lessons you are unable to get the demos working, or if you want to view the demos first before reading the lessons, you can install the Inventory Tracker demo package with pre-configured connectors. Once installed, you can start viewing the demos right away.

Before installing the Inventory Tracker demo with pre-configured connectors, you need to remove any new connectors that you may have created while following the tutorials.

### ❖ To remove connectors

1. Choose Start ► Programs ► RFID Anywhere ► Administrator Console.  
The Administrator Console opens.
2. At the Login page, enter your credentials and click Log On.  
The Home page appears.
3. If you have followed the lessons in this chapter and have created some or all of the connectors, you need to remove them first. Select the following connectors and then click Remove.
  - ◆ InvTracSim1
  - ◆ InvTracSim2
  - ◆ InvTracBMTCP
  - ◆ InvTracAleTCP
  - ◆ InvTracSource1ALE
  - ◆ InvTracSource2ALE
  - ◆ InvTracCountALE
  - ◆ MPReport
  - ◆ FileOutput

If you installed the Inventory Tracker business module, then you need to uninstall it. If you deployed the Inventory Tracker business module from Visual Studio, then you need to uninstall the business module by deleting the DLL file from the *bms* folder. If you installed the Inventory Tracker business module using Provisioning, then you can use Provisioning to uninstall it.

### ❖ To uninstall the Inventory Tracker business module when deployed from Visual Studio

1. Close the Administrator Console.
2. From the Start menu, choose Start ► Programs ► RFID Anywhere ► Stop Service.  
The RFID Anywhere service stops.
3. Go to the *bms* folder located in the RFID Anywhere installation directory. The default location is *C:\Program Files\Sybase\RFID Anywhere\bms*.
4. Right-click the *AA6B44F6-9DB6-4961-9DC1-D4D2F6788D0C* folder and choose Delete. Click Yes to confirm the deletion.

When a property of a business module (or a connector) is modified from the Administrator Console Properties Manager, a *.rn* file is create in the *data/[GUID]* folder of your RFID Anywhere installation. The *.rn* file stores the values of the properties exposed in the Properties Manager pane. Once you have uninstalled the Inventory Tracker business module, you need to delete the *InventoryTracker.rn* file. Use the following delete the file.

### ❖ To uninstall the Inventory Tracker business module using Provisioning

1. Click the Provisioning tab.
2. Select InventoryTrackerBM in the Packages list.
3. Click Remove in the Packages menu.

The package is first uninstalled and then removed from the list.

Note that you can also click Uninstall first and then Remove. Uninstall will uninstall the business module and store the MSI file under the *Packages\Repository* folder of your RFID Anywhere installation directory. The status of the business module under the Packages list will change to Stored. You can then click Remove to completely remove the business module.

4. It is important to note that when a package is removed, the RFID Anywhere service gets restarted. Once the RFID Anywhere service has restarted, click Refresh to view the updated list.

For all business modules, whether they were deployed using Visual Studio or provisioning, when business module (connector) property is modified from the Administrator Console Properties Manager, a *.rn* file is created in the *data/[GUID]* folder of your RFID Anywhere installation. The *.rn* file stores the values of the properties exposed in the Properties Manager pane. Once

you have uninstalled the Inventory Tracker business module, you also need to delete the *InventoryTracker.rm* file. Follow the procedure *To clear Inventory Tracker business module properties* (below) to delete the file.

❖ **To clear Inventory Tracker business module properties**

1. Close the Administrator Console.
2. From the Start menu, choose Start ► Programs ► RFID Anywhere ► Stop Service.  
The RFID Anywhere service stops.
3. Go to the *data\AA6B44F6-9DB6-4961-9DC1-D4D2F6788D0C* folder located in the RFID Anywhere installation directory. The default location is *C:\Program Files\Sybase\RFID Anywhere\data\AA6B44F6-9DB6-4961-9DC1-D4D2F6788D0C*.
4. Right-click the *InventoryTracker.rm* file and choose Delete. Click Yes to confirm the deletion.

Now that you have uninstalled all of the previously created components and have cleared the properties of the Inventory Tracker business module, you can install the pre-configured Inventory Tracker package.

❖ **To install the pre-configured Inventory Tracker package**

1. Close the Administrator Console.
2. From the Start menu, choose Start ► Programs ► RFID Anywhere ► Stop Service.  
The RFID Anywhere service stops.
3. Open the *InventoryTrackerSetup.msi* file. This file is located in the *Inventory Tracker\PreConfigMSI* folder of your RFID Anywhere installation directory.  
The Inventory Tracker setup opens.
4. Click Next to proceed with the installation.
5. Click Next to start the installation.
6. Click Close to exit Inventory Tracker setup.
7. From the Start menu, choose Start ► Programs ► RFID Anywhere ► Start Service.  
The RFID Anywhere service starts.


8. Open the Administrator Console and login to the Home page.  
All of the Inventory Tracker connectors are loaded in the Local Services list.

❖ **To start the Inventory Tracker demo components**

1. Open the Administrator Console Home page.
2. From the Local Services list, select the `MPRServiceModule` and then click Edit.
3. Expand the Logical Readers property. Ensure that the entry **`MPSource:InvTracSim1`** exists.  
Enter the above value if it is not already present.  
Click the green disk in the title bar of the Properties Manager to save any changes.
4. From the Local Services list, select the `ALEServiceModule` and then click Edit.
5. Expand the Logical Readers property. Ensure that the entries **`Source1:InvTracSim1`** and **`Source2:InvTracSim2`** exist.  
Enter the above values if they are not already present.  
Click the green disk in the title bar of the Properties Manager to save any changes.
6. From the Local Services list, select the following connectors and then click Restart.
  - ◆ `MPRServiceModule`
  - ◆ `InventoryTracker`
  - ◆ `InvTracSim1`
  - ◆ `InvTracSim2`
  - ◆ `InvTracBMTCP`
  - ◆ `InvTracAleTCP`
  - ◆ `ALEServiceModule`
  - ◆ `InvTracSource1ALE`
  - ◆ `InvTracSource2ALE`
  - ◆ `InvTracCountALE`
  - ◆ `MPReport`
  - ◆ `FileOutput`

7. Double-click *ALEDemo.exe*, *EnterpriseApp.exe* and *WebServicesApp.exe*.

The shortcuts to these files are located in the *Inventory Tracker* folder of your RFID Anywhere installation directory.

 For instructions on running the demo applications, refer to the following sections in this chapter.

- ◆ To run the ALE demo application which generates and processes ALE event cycle reports, see [“Starting the ALE demonstration” on page 214](#).
- ◆ To run the Report Engine MP demo, see [“Starting the Report Engine MP demo” on page 219](#).
- ◆ To run the business module demo which shows how an application processes business module output, see [“Starting the business module demonstration” on page 224](#).
- ◆ To run the web services demonstration which shows how an application can remotely call business module methods, see [“Starting the web services demonstration” on page 226](#).

## Summary

This chapter introduced you to the Inventory Tracker demo and its components. The chapter also introduced you to custom business modules and provided examples of the various RFID Anywhere development models: Application Level Events (ALE), Report Engine MP, and custom business modules.

Unlike ALE or Report Engine MP, business modules can apply business logic and add business value prior to sending data to an enterprise application. They are used to process complex business logic at the edge.

☞ To learn more about custom business module development and how to create your own business modules, see [“Using the RFID Anywhere Visual Studio Extension to Create Custom Business Modules”](#) [*RFID Anywhere Developer’s Guide*, page 7].



---

# Index

## A

accessing managed systems	40	about	155
adding and configuring connectors	104	ALE Report Viewer	
adding managed systems		tutorial	169
about	40	ALE Web Service module	
adding tag groups	83	about	158
administering		Application Level Events (ALE)	
RFID networks	39	about	8
Administrator Console		ASP.NET	
about	31	installing the .NET framework	18
accessing managed systems	40	<b>B</b>	
adding local services	35	business modules	
adding managed systems	40	about	7
administering multiple systems	39	deploying packages	47
Configuration page	51	package versioning	48
configuring local services	38	provisioning	47
Domain Security page	55	viewing properties	38
editing local services	36	<b>C</b>	
Home page	34	clearing an event	45
Local Security page	53	Component Manager	
Login page	33	installing	20
Properties Manager	38	Configuration page	
Provisioning page	47	Administrator Console	51
refreshing local services list	37	configuring local services	38
removing events	45	configuring tag groups	85
removing local services	38	connector properties	
removing managed systems	41	File messaging	106
restarting local services	37	MSMQ messaging	110
starting	32	QAnywhere messaging	113
starting local services	36	SMTP messaging	118
stopping local services	37	TCP messaging	121
using	32	UDP messaging	124
administrator console		Web service messaging	127
about	10	connectors	
ALE		about	6
about	154	adding messaging	104
RFID Anywhere implementation	155	adding messaging connectors to error	
tutorial	153	notifications	45
ALE basics	154	ALE event cycle	155
ALE Event Cycle connectors		File messaging	106
properties	176	installing	105
ALE event cycle connectors			

messaging	102	data format	97
MSMQ messaging	110	Decode method	96
MultiProtocol Report Definition properties	194	Encode method	96
QAnywhere messaging	113	encryption	97
SMTP messaging	118	protocol selection	97
TCP messaging	121	schema	96
UDP messaging	124		
viewing properties	38		
Web Service messaging	127		
controllers			
about	6		
creating simulated RFID data	137		
creating simulator files			
about	82		
Creation Mode pane			
RFID Simulator Data Editor	80		
<b>D</b>		<b>E</b>	
data formats		ECSpec parameter	
DPP	97	about	157
data formats encryption		editing domain security	55
DPP	97	editing local security	53
data protocol processors		editing local service properties	36
about	96	editing local services	36
Decode method		Encode method	
DPP	96	DPP	96
default local services	58	Encoding pane	
default RFID Anywhere services	57	RFID Simulator Data Editor	81
deleting packages	50	EPC tags	
deleting tag groups	85	simulating	80
demos		error logs	
Inventory Tracker	203	deleting	44
deployment		deleting contents	45
adding components to an RFID Anywhere		updating	44
network	47	viewing	43, 44
package versioning	48	errors	
packages	47	receiving notification	45
scheduling	48	event cycle connectors	
documentation		about	155
providing feedback	ix	events	
RFID Anywhere	viii	removing from Administrator Console	45
domain security		removing from the Events list	44
editing	55	updating log files	44
Domain Security page		viewing for local services	43
Administrator Console	55	viewing log file information	44
DPP		Events list	
about	9, 96	about	43
		refreshing contents	44
		removing events	44
		<b>F</b>	
		feedback	
		documentation	ix
		providing	ix
		File messaging connector	
		description	106
		properties	106

requirements	106	Label Designer	
using for troubleshooting	130	about	11, 89
<b>G</b>		Label Canvas pane description	90
getting started with RFID Anywhere	3	Properties pane description	90
Group pane		starting	90
RFID Simulator Data Editor	80	using	90
<b>H</b>		Windows Forms list description	90
hardware		Label Designer files	
using with Inventory Tracker demo	229	about	91
Home page		creating	91
Administrator Console	34	opening	92
<b>I</b>		saving	91
IIS		using	93
required by RFID Anywhere	17	Label Designer output files	
immediate operational mode		about	91
about	156	license keys	
installing		updating	42
IIS	17	licensing	
MSMQ	16	RFID Anywhere	42
.NET framework	18	updating license keys	42
RFID Anywhere	15	local security	
WSE	19	about	53
installing future connectors	105	Local Security page	
installing RFID Anywhere	20	Administrator Console	53
installing the RFID Anywhere Visual Studio		local services	
Extension	23	about	34
Internet Explorer		adding	35
required for Administrator Console	32	configuring	38
Inventory Tracker Demo		deleting events	45
using	203	editing properties	36
Inventory Tracker demo		installed by default	58
adding real hardware	229	managing	34
ALE demo	211	provisioning	47
business module demo	221	refreshing information	37
customizing	228	removing	38
installing pre-configured version	232	restarting	37
Report Engine MP demo	216	starting	36
using web services to access business module		stopping	37
methods	226	types supported by RFID Anywhere	35
<b>L</b>		viewing events	43
Label Canvas pane		viewing log file information	44
Label Designer	90	log files	
		deleting	44
		updating	44
		viewing contents	44
		Login page	
		Administrator Console	33

**M**

Managed Systems 39  
 about 39  
 managed systems  
 accessing 40  
 adding 40  
 removing 41  
 managing local services 34  
 message queuing  
 installing MSMQ 16  
 messaging connectors  
 about 9, 102  
 adding 104  
 installing 105  
 receiving error notifications 45  
 troubleshooting 130  
 modes  
 choosing between site manager and component manager 20  
 monitoring  
 about 43  
 removing events 44  
 monitoring an RFID Anywhere network 43  
 Monitoring page  
 using for troubleshooting 131  
 .msi files  
 about 47  
 MSMQ  
 required by RFID Anywhere 16  
 MSMQ messaging connector  
 about 110  
 properties 110  
 requirements 110  
 MultiProtocol Report Definition connectors  
 properties 194  
 multiprotocol reports  
 sample 191

**N**

.NET Framework  
 required by RFID Anywhere 18  
 networks  
 monitoring RFID Anywhere 43

**O**

obtaining system information 44

opening Label Designer files  
 about 92  
 opening simulator files  
 about 83

**P**

package versioning 48  
 packages  
 about 47  
 deleting 50  
 deploying 47  
 scheduling deployment 48  
 uninstalling 49  
 updating status 49  
 using provisioning 47  
 versioning 48  
 poll operational mode  
 about 156  
 properties  
 ALE Event Cycle connectors 176  
 configuring local services 36  
 editing for business modules 38  
 editing for connectors 38  
 MultiProtocol Report Definition connectors 194  
 obtaining descriptions and settings 39  
 setting for tag groups 85  
 Properties Manager  
 about 34  
 using 38  
 viewing property descriptions 39  
 Properties pane  
 Label Designer 90  
 protocols  
 DPP 97  
 provisioning  
 about 47  
 refreshing packages 49

**Q**

QAnywhere messaging connector  
 about 113  
 properties 113  
 requirements 113

**R**

radio frequency identification *see* RFID

refreshing local services	37	creating simulated	137
refreshing packages	49	simulating	80
refreshing the events list	44	RFID Network Simulator	
removing local services	38	about	7
removing managed systems	41	RFID networks	
removing packages	50	administering	39
Report Engine MP		RFID Simulator Data Editor	
about	183	about	79
report engine MP		adding tag groups	83
about	8	configuring tag groups	85
restarting local services	37	Creation Mode pane description	80
RFID Anywhere		deleting tag groups	85
about	4	Encoding pane description	81
ALE implementation	155	Group pane description	80
documentation	viii	starting	80
installation steps	20	Tag Groups list description	80
installing	15	Tags pane description	81
installing Visual Studio Extension	23	using	80
licensing	42	using simulator files	82
product description	4	RFID tags	
starting	25	simulating	83
system requirements	15	<b>S</b>	
RFID Anywhere ALE Report Viewer		saving Label Designer files	
tutorial	169	about	91
RFID Anywhere Getting Started Guide	vii	saving simulator files	
RFID Anywhere messaging connectors	101	about	83
RFID Anywhere networks		scheduling packages for deployment	48
accessing managed systems	40	schemas	
adding local services	35	DPP	96
adding managed systems	40	security	
configuring local services	38	domain	55
editing local services	36	local	53
managing local services	34	services	
monitoring	43	adding local services to an RFID Anywhere	
obtaining system information	44	network	35
refreshing local services list	37	configuring local services	38
removing local services	38	editing local	36
removing managed systems	41	local	58
restarting local services	37	managing local services	34
simulating	80	refreshing local services list	37
starting local services	36	removing local services	38
stopping local services	37	required	23
RFID Anywhere service		restarting local services	37
starting	25	starting local	36
RFID applications		starting the RFID Anywhere service	25
testing	80	stopping local services	37
RFID data			

Services Manager		RFID Simulator Data Editor	81
about	34	TCP messaging connector	
using	34	description	121
simulated data		properties	121
creating	137	requirements	121
simulating RFID Anywhere networks		testing	
RFID Simulator Data Editor	80	RFID applications	80
simulator files		troubleshooting	
about	82	messaging connectors	130
creating	82	using File messaging connector	130
opening	83	using the Monitoring page	131
saving	83	tutorials	
Site Manager		creating simulated RFID data	137
installing	20	descriptions	135
managed systems	39	RFID Anywhere ALE Report Viewer	169
SMTP messaging connector		using ALE to filter and view RFID data	153
about	118	using Report Engine MP	183
properties	118	using the Inventory Tracker Demo	203
requirements	118		
starting local services	36	<b>U</b>	
starting the Administrator Console	32	UDP messaging connector	
starting the Label Designer	90	description	124
starting the RFID Simulator Data Editor	80	properties	124
status		requirements	124
updating package status	49	uninstalling packages	49
stopping local services	37	updating a license key	42
subscribe operational mode		using ALE to filter and view RFID data	153
about	157	using provisioning	47
System Monitoring list		using Report Engine MP	183
about	43	using the Administrator Console	32
system requirements		using the Label Designer	90
RFID Anywhere	15	using the Properties Manager	38
		using the Properties Manager help	39
<b>T</b>		using the RFID Simulator Data Editor	80
tag groups			
about	80	<b>V</b>	
configuring	85	versioning	
deleting	85	packages	48
simulating RFID data	83	Visual Studio Extension	
Tag Groups list		about	10
RFID Simulator Data Editor	80	installing	23
tags		Visual Studio .NET	
configuring simulation properties	85	installing the RFID Anywhere Extension	23
creating simulator tag groups	83		
deleting tag groups	85	<b>W</b>	
simulating RFID networks	80	web server	
Tags pane			

installing IIS	17
Web Service messaging connector	
description	127
properties	127
requirements	127
web services	
installing WSE	19
Windows Forms list	
Label Designer	90
Windows services	
required for RFID Anywhere	23
working with Label Designer output files	91
working with simulator files	82
WSE	
required by RFID Anywhere	19