



RFID Anywhere™ Mobile Support Guide

September 2008

Version 3.5.1

Copyright and trademarks

Copyright © 2008 iAnywhere Solutions, Inc. Portions copyright © 2008 Sybase, Inc. All rights reserved.

This documentation is provided AS IS, without warranty or liability of any kind (unless provided by a separate written agreement between you and iAnywhere).

You may use, print, reproduce, and distribute this documentation (in whole or in part) subject to the following conditions: 1) you must retain this and all other proprietary notices, on all copies of the documentation or portions thereof, 2) you may not modify the documentation, 3) you may not do anything to indicate that you or anyone other than iAnywhere is the author or source of the documentation.

iAnywhere®, Sybase®, and the marks listed at <http://www.sybase.com/detail?id=1011207> are trademarks of Sybase, Inc. or its subsidiaries. ® indicates registration in the United States of America.

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

Contents

- About this book v**
 - RFID Anywhere documentation vi
 - Finding out more and providing feedback vii

- I. RFID Anywhere Mobile Support Overview 1**
 - 1. Understanding RFID Anywhere's mobile device support 3**
 - Mobile device support overview 4
 - Mobile device support components 5

- II. RFID Anywhere Mobile Device Connectors 7**
 - 2. RFID Anywhere mobile device connectors 9**
 - Architecture overview 10
 - Installing connectors 11

- III. Developing Mobile Applications 13**
 - 3. Developing RFID Anywhere applications for mobile devices 15**
 - Introduction to developing RFID Anywhere applications for mobile devices 16
 - Option 1: Using the RFID Anywhere mobile agent 17
 - Option 2: Building custom applications on device 18

- Index 25**

About this book

Subject

This book outlines RFID Anywhere's mobile device support and development model.

Audience

This manual is for developers using RFID Anywhere's mobile device support for handheld RFID readers, barcode scanners, and forklift readers.

Before you begin

This book assumes a familiarity with RFID Anywhere and its development options and configuration/management tasks.

RFID Anywhere documentation

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

The RFID Anywhere documentation set consists of the following components:

- **RFID Anywhere Getting Started Guide** This book 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 book introduces common development tasks often undertaken when working with RFID Anywhere. It describes the RFID Anywhere Visual Studio Extension for creating custom business modules, introduces commonly-used interfaces for hardware and component interaction, and outlines how to write Data Protocol Processors (DPPs).
- **RFID Anywhere Enterprise Manager Guide** This book describes RFID Anywhere Enterprise Manager, a standalone management tool providing dynamic, standards-based management for distributed RFID Anywhere networks.
- **RFID Anywhere Mobile Support Guide** This manual outlines RFID Anywhere's mobile device support and development model.
- **RFID Anywhere Location Information System Guide** This book describes RFID Anywhere's Location Information System (LIS). Using RFID Anywhere and LIS, businesses can not only track assets from a range of data collection points [such as passive RFID, active RFID, real-time location systems (RTLS), barcode, global positioning systems (GPS) and environmental sensors], but can dramatically enhance the value of this data by associating environmental inputs (e.g., temperature) along with the location data. In addition, the RFID Anywhere Location Information System exposes its complete feature set via a rich service oriented architecture (SOA) web services layer allowing organizations to easily transform static business information into business-aware intelligence.
- **RFID Anywhere Label Designer Guide** This book describes RFID Anywhere's printing capabilities and its Label Designer, a graphical tool used to design the layout of RFID and barcode labels.

Visit 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.sybase.com/products/rfidsoftware/rfidanywhere>.

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. Although we do not reply to emails sent to this address, we read all suggestions with interest.

Part I. RFID Anywhere Mobile Support Overview

This part provides an introduction to RFID Anywhere's mobile device support.

CHAPTER 1

Understanding RFID Anywhere's mobile device support

Contents

Mobile device support overview 4
Mobile device support components 5

Mobile device support overview

This chapter outlines RFID Anywhere's support for mobile devices, including handheld RFID readers, handheld barcode scanners, and forklift RFID readers built on Windows CE .NET. For detailed information about developing mobile applications with RFID Anywhere's mobile device support, see [“Developing RFID Anywhere applications for mobile devices” on page 15](#).

RFID Anywhere's mobile device support integrates mobile devices into the RFID Anywhere framework. Mobile devices require special support and need to be looked at as a unique device family, as their functionality and usage differs greatly from fixed RFID readers and barcode scanners. By integrating these devices into RFID Anywhere, developers can leverage RFID Anywhere's business module architecture to allow the flexible handling of *events* based on the data captured by the device.

RFID Anywhere's mobile device support provides two unique options for application development and integration.

In the first model, an on-device agent can send collected data to RFID Anywhere automatically with no device-side programming. These events can then be handled by a custom business module.

In the second model, easy-to-use APIs in RFID Anywhere's mobile libraries on the device allow application developers to build .NET Compact Framework applications that run on the device and function as extensions to RFID Anywhere.

In both cases, data sent between a mobile device and an RFID Anywhere server is sent *asynchronously*. Data collected while the mobile device has no network connectivity is routed to the RFID Anywhere site when connectivity is restored. Developers do not have to program to a messaging system on the device, as the RFID Anywhere mobile libraries provide the asynchronous layer. Thus, developers can easily create applications that can work online or offline, without having to do the underlying configuration or programming themselves.

Mobile device support components

This section outlines the various components of RFID Anywhere's mobile device support.

RFID Anywhere mobile agent

The RFID Anywhere mobile agent is an out-of-the-box interface that automatically routes data (RFID and barcode reads) from the mobile device to an RFID Anywhere server, with no device-side programming required.

RFID Anywhere mobile device libraries

The RFID Anywhere mobile device libraries are .NET Compact Framework libraries that are installed on the mobile device, enabling developers to write custom, device-agnostic applications. These libraries provide asynchronous communication without requiring developers to write any underlying communication code. Developers can also programmatically interact with reader modules (RFID, barcode) on the device through the API exposed through the RFID Anywhere mobile device libraries.

Controller for mobile devices

A controller for mobile devices is installed on the RFID Anywhere server. This controller provides the server-side abstraction and common API for all mobile devices supported by RFID Anywhere.

Device-specific connectors

Connectors developed by iAnywhere engineering to support specific mobile devices provide native integration to and configuration of specific mobile devices. By using connectors, developers can write device-agnostic code and not worry about low-level communication with the mobile device.

Business module framework

The business module framework provided by RFID Anywhere handles events from all types of hardware to enable flexible process automation and integration. It exposes a powerful server-side framework to allow real-time event handling of events from mobile devices.

Other Sybase iAnywhere technologies

Sybase iAnywhere is recognized as a leader in enterprise mobility. Our product portfolio of development tools, databases, and management tools can help make your mobile projects a success. For more information, visit www.sybase.com/ianywhere.

Part II. RFID Anywhere Mobile Device Connectors

This part outlines the various components of an RFID Anywhere connector for a mobile device, including installation tips and requirements.

CHAPTER 2

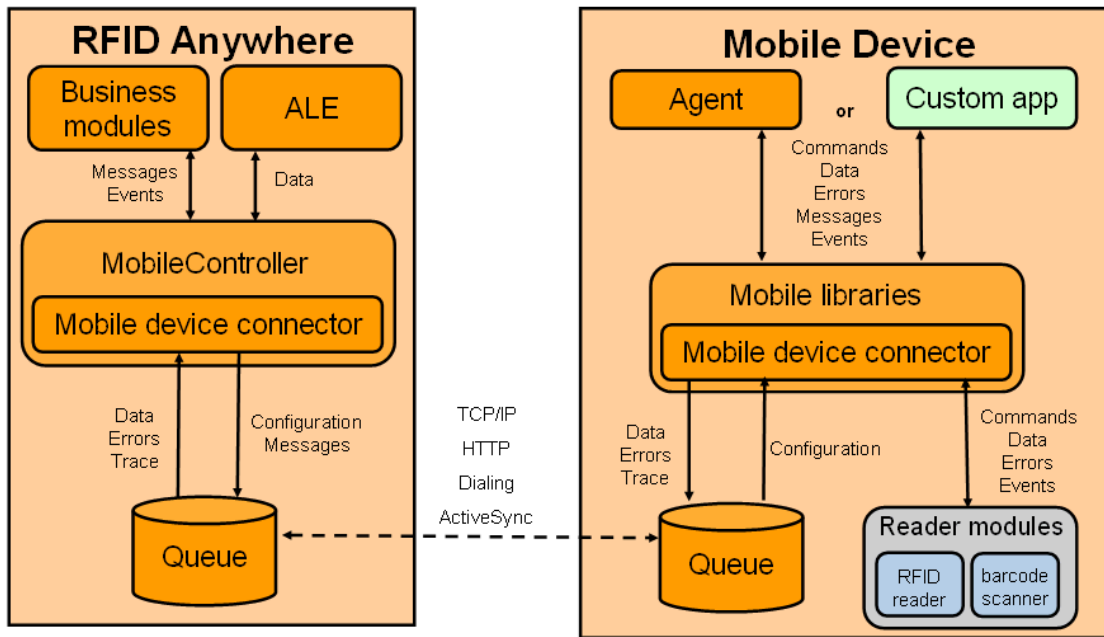
RFID Anywhere mobile device connectors

Contents

Architecture overview 10
Installing connectors 11

Architecture overview

RFID Anywhere's mobile device support architecture includes connectors that abstract the low-level device interaction by providing a standard API and asynchronous communication layer across a variety of devices, making it possible to build device-agnostic, mobile RFID applications. Communication of data, events, messages, and configuration between the device and the server is handled automatically using asynchronous messaging.



Installing connectors

The server-side device connector is installed in the same manner as connectors to fixed RFID readers or RFID printers from the Provisioning page of the RFID Anywhere Administrator Console. After installing the server-side support, the communication between the server and device must be verified, and the device-side connector must be installed to the device.

Since the server-side mobile device connector and the device-side mobile connector communicate with each other asynchronously using Web standards, the network communication must be configured correctly. Port 80, for HTTP traffic, must be open on the RFID Anywhere server for the mobile device to communicate with it.

You can test to make sure this ports are open by running the following command from a command prompt from a computer in the same domain or workgroup as both the RFID Anywhere server and the mobile device:

```
telnet IP-address-of-RA-server 80
```

where *IP-address-of-RA-server* is the IP address of the RFID Anywhere server.

In addition, the mobile device and RFID Anywhere server must be in the same domain or workgroup for the web-based asynchronous messaging to function correctly. Alternatively, ActiveSync can be used to facilitate the server-device communication.

Note

If the asynchronous messaging between the device and server cannot successfully communicate with one another, no data is transferred and the RFID Anywhere mobile support does not function.

To install the device-side connector to the device

1. Deploy the connector package for the device to the RFID Anywhere server using Provisioning.
See “Deploying packages” [*RFID Anywhere Getting Started Guide*].
2. Launch Pocket Internet Explorer from the device and navigate to **http://IP-address-of-RA-server/rfidanywhere/mobile**, where *IP-address-of-RA-server* is the IP address of the RFID Anywhere server.
3. From the list of devices, select the appropriate device to download the CAB file that installs the connector and agent files to the device.
4. When prompted, choose to execute the downloaded CAB file.

Note

This deployment step also verifies that the device can communicate to the server via HTTP.

To verify the server/device communications using the RfidNet Agent

1. Configure the device's host name from the server-side connector. Set the connector's Trace Level property to Verbose. Start the connector.

Note

The device's host name, or unique device name, can be found using tools on the device, using a router's management capabilities, or from the device's **System** page.

2. From the mobile device, use the File Explorer application to navigate to *\Program Files* and launch the program *rnagent*.
3. Click **Connect** to connect to the RFID Anywhere mobile system, and enter the host name or IP address of the RFID Anywhere server when prompted.
4. Scan barcodes and RFID tags. These appear on the screen, and are sent to the RFID Anywhere server.
5. From the **Monitoring** page of the RFID Anywhere Administrator Console, select the connector from the dropdown list, and scan the log entries for the RFID and barcode data.

If the data shows up, the RFID Anywhere server, mobile device, and communication layer are configured correctly.

Part III. Developing Mobile Applications

This part provides detailed instructions and code samples for developing mobile applications with RFID Anywhere.

CHAPTER 3

Developing RFID Anywhere applications for mobile devices

Contents

Introduction to developing RFID Anywhere applications for mobile devices 16

Option 1: Using the RFID Anywhere mobile agent 17

Option 2: Building custom applications on device 18

Introduction to developing RFID Anywhere applications for mobile devices

This chapter builds on the introductory mobile device support content from [“Understanding RFID Anywhere's mobile device support”](#) on page 3.

It also provides steps and instructions for creating applications with RFID Anywhere that integrate mobile devices. RFID Anywhere's mobile device support provides a number of options for integrating mobile devices into your RFID network.

When developing a mobile application with RFID Anywhere's mobile device support, the following steps must be taken:

1. Connect to the RFID Anywhere mobile system
2. Register to receive events
3. Detect RFID notification events
4. Process events (RFID, barcode, application messages) as they are fired
5. Disconnect from the RFID Anywhere mobile system

Option 1: Using the RFID Anywhere mobile agent

RFID Anywhere's mobile device support includes a pre-built, on-device agent that implements the API of the RFID Anywhere mobile libraries. This agent sends all collected RFID tag and barcode data to RFID Anywhere with no device-side programming. With this model, there is no complex business logic on the device. Developers instead build all business logic and data processing at the RFID Anywhere server in the form of business modules.

Under the covers, all RFID reads and barcode scans are received by the mobile agent and sent using asynchronous messaging to the RFID Anywhere server. If the device is working offline, the data is stored on the device until the device connects again, and is sent.

When data is sent from the mobile device to RFID Anywhere, it is picked up by the associated hardware connector, and RFID Anywhere's Mobile Controller exposes the data as events. These events are then sent to any business module that has subscribed to data from the specific connector, and these events are handled by business logic. The Mobile Controller generates `RfidMPEventArgs` for RFID data (just like the RFID Multiprotocol Controller) and `BarcodeEventArgs` (just like the Barcode Controller).

Developers can write business modules to handle RFID and barcode events as they are generated and sent from the device. These events are handled in the same manner as events from the multiprotocol RFID reader controller and the barcode controller.

Note

One difference to be noted is that since handheld RFID readers do not have the notion of smoothing as fixed readers do, all `RfidMPEventArgs` generated for RFID events have a value of `TagEventType.New` for the `EventType` property.

For more information about working with events from the Mobile controller within a business module, see [“Mobile controller” \[RFID Anywhere Developer's Guide\]](#).

In addition, the data from mobile devices can also be used to populate Application-level Event reports by configuring the mobile device as a data source for the report. See [“Using ALE to filter and view RFID Data” \[RFID Anywhere Getting Started Guide\]](#).

The Mobile Controller's default business module, named `MobileModule`, also exposes two properties that allow events to be automatically sent to enterprise systems without the need for custom business module development. The property `Tag Notification` can be set to the name of a configured messaging connector, and when RFID events are generated by the Mobile Controller, these are sent according to the messaging connector's details. Similarly, the `Barcode Notification` property can be set to the name of a configured messaging connector, and when barcode events are generated by the Mobile Controller, these are sent according to the messaging connector's details.

Option 2: Building custom applications on device

In many RFID implementations that incorporate mobile devices, it is useful to have business logic and feedback mechanisms running locally on the mobile device. RFID Anywhere's mobile device support allows application developers to develop .NET Compact Framework applications that run on the device and function as extensions to RFID Anywhere. The mobile device support includes easy-to-use APIs for sending and receiving custom application data to and from RFID Anywhere business modules, and for interacting with device hardware. These libraries expose RFID reads and barcode scans as events that can be trapped and handled by a custom application on the device. From there, an application could process the raw event by applying some business logic, or it could display the details of the event to the user as a feedback mechanism.

By using RFID Anywhere's mobile libraries, applications can run on different devices, since these libraries and the device connector layer abstract device specific interfaces.

Custom applications can also communicate easily with RFID Anywhere using application messages. Devices can receive messages from RFID Anywhere, and can send messages to RFID Anywhere as well. These messages are most commonly sent from and processed by custom business modules running in RFID Anywhere. See [“Application message events from Mobile controller” \[RFID Anywhere Developer's Guide\]](#).

When working with application messages, asynchronous queuing is used under the covers. By abstracting this layer in the RFID Anywhere mobile libraries, developers do not have to build this offline, queuing infrastructure themselves, allowing them to focus on message processing, device interaction, and other on-device business logic.

Note

Even when application messages are used as the communication layer between the mobile device and RFID Anywhere, all RFID reads and barcode scans are still automatically sent to RFID Anywhere, similar to when only the mobile agent is used.

Mobile application developers can also take advantage of RFID Anywhere's Data Protocol Processor (DPP) architecture to encode and decode tags programmatically. This is another unique tool for developers that allows the creation of powerful on-device applications. See [“Getting started with Data Protocol Processors \(DPPs\)” \[RFID Anywhere Getting Started Guide\]](#).

Mobile library API code samples

This section includes a variety of .NET Compact Framework code samples to showcase the simplicity of the device-agnostic, RFID Anywhere mobile libraries. Sample applications are also provided as examples of how mobile projects can be created and deployed with Visual Studio. These code samples cover applications that leverage RFID Anywhere's RFID, barcode, and application messaging support.

Connecting to the RFID Anywhere mobile system

Every mobile RFID Anywhere application requires an RfidNetManger object. This object is the RFID Anywhere's mobile device manager, and it allows the developer to control RFID triggers, barcode scans, and to transfer custom application messages with the RFID Anywhere server. The following code sample shows the recommended method of setting up the RfidNetManager and using it to connect to the device's RFID reader and barcode scanner.

```
private RfidNetManager mgr;

//Create an instance of the RFID Net Manager.
mgr = RfidNetManager.GetInstance(true);

try{
//Connect the manager. This starts the RFID Anywhere system.
mgr.Connect();

// Load the devices
IDevice barcodeReader =
(IBarcodeReader)mgr.LoadDevice(typeof(IBarcodeReader));
IDevice rfidReader = (IRfidReader)mgr.LoadDevice(typeof(IRfidReader));
return true;
}
catch(Exception e){
/**Handle errors here**/
return false;
}
}
```

Registering to receive events

Once the RfidNetManager is connected, the application needs to be set up to listen for RFID, Barcode, and Application Messaging events. The following code sample demonstrates how to register for RFID data events. Here, the developer-created rfidReader_DataHandler method is used to handle RFID data events.

```
// register event handler for RFID data
rfidReader.OnData += new OnDataEventHandler(this.rfidReader_DataHandler);
```

By providing an event handling architecture, RFID Anywhere allows developers complete control and flexibility as to what to do with RFID, barcode, and application message data in their application.

Processing RFID data

When tags are read, the OnData event is generated and the handler is called. The GetRfidTagList method can be used to retrieve the tags read by the reader. Developers can then perform tasks such as building custom output, adding the tag to a database, decoding the tag data, sending output to the server as an application message, displaying data in a GUI, filtering, or any other functionality that is required. The general form of a handler method is as follows.

```
/**
 * Method for handling RFID Anywhere Data events.
 * */
private void rfidReader_DataHandler(object sender, DataEventArgs e)
{
```

```
// Check for an Identify Event signaling that data is available
if (e.EventType == DataEventType.Identify)
{
    RFIDTag[] tags = (IRfidReader)sender.GetRfidTagList();
    if (tags != null)
    {
        foreach(RFIDTag tag in tags)
        {
            // decode, write to local DB, send message, etc.
        }
    }
}
```

Detecting RFID notification events

The following RFID notification events need to be detected and handled:

- ConfigurationApplied
- Error
- IndentifyStarted
- ReadCompleted
- TriggerPulled
- TriggerReleased
- Unknown
- WriteCompleted

The event that all RFID applications should handle is TriggerPulled. When this event occurs, the RFID system needs to activate the RFID reader. The following code sample demonstrates this.

```
// specify handler for RFID notification events
rfidReader.OnNotify +=new
OnNotifyEventHandler(this.rfidReader_NotificationHandler);

...

/**
 * This is the local method for handling RFID Notification
 * events. This method detects different types of RFID events and
 * performs some type of action on them.
 * */
private void rfidReader_NotificationHandler(object sender, NotifyEventArgs e)
{
    if (sender is IRfidReader)
    {
        if (e.Notification == NotificationType.TriggerPulled)
        {
            // Received RFID TriggerPulled event
            // Tell the reader to start reading the tag IDs.
            try
            {
                ((IRfidReader)sender).Identify();
            }
            catch
            {
                // handle errors here
            }
        }
    }
}
```

Note

There are no barcode or application messaging notification events that need to be detected.

Processing and detecting barcode data

When barcodes are read, the system automatically calls the method that was registered when the barcode was started. Barcodes are read one at a time.

When barcodes are read, the OnData event is fired. The GetBarcodeList method can be used to retrieve the tags read by the reader. Developers can then perform tasks such as building custom output, adding the tag to a database, sending it to server as an application message, displaying data in a GUI, filtering, or any other situation that is required. The general form of an OnData method is below.

```
// specify handler for barcode data events
barcodeReader.OnData += new
OnDataEventHandler(this.barcodeReader_DataHandler);

..

/**
 * Method for handling barcode data events.
 * This method handles the barcode data.
 * */
private void barcodeReader_DataHandler(object sender, DataEventArgs e)
{
    try
    {
        // Check for a 'Read' Event signaling that data is
        available.
        if (e.EventType == DataEventType.Read)
        {
            //Variable to store an array of barcodes.
            BarcodeData[] barcodes =
            ((IBarcodeReader)sender).GetBarcodeList();
            if (barcodes != null && barcodes.Length > 0 )
            {
                // Barcodes only read one at a time
                BarcodeData barcode = barcodes[0];
                // Enter additional custom logic here

            }
        }
    }
    catch(Exception error)
    {
        // error handling
    }
}
```

Detecting application messages sent from the server

To detect a custom message from the server to the mobile device, the following code is required.

```
mgr.OnMessageReceived +=new OnMessageReceivedHandler(AppMessageHandler);
```

```
...
/**
 * Perform some function on a received message.
 * */
private void AppMessageHandler(object sender, AppMessageEventArgs e)
{
    // Receiving message from server
    if (e.Message != null)
    {
        myOutput = e.Message.Name + " : " + e.Message.MessageData +
" : " + e.Message.Argument + " : " + System.DateTime.Now.ToString();
    }
}
```

For more information about sending application messages from the RFID Anywhere server, often from a business module or web services application, see [“Mobile controller” \[RFID Anywhere Developer's Guide\]](#).

Sending application messages from the device to the server

To send a custom message to the server, you must have:

- A valid connection to the RfidNetManager (as shown above)
- A custom message of type `iAnywhere.RfidNet.Mobile.ApplicationMessage`

The following code demonstrates how to setup a valid `iAnywhere.RfidNet.Mobile.ApplicationMessage` object and send it to the server.

```
/**
 * Method to handle custom messages.
 * Requires: String - Message Data, String - Message Argument (no
 * spaces allowed), String - Message Name (no spaces allowed)
 * */
private bool CustomMsg(string messageData, string messageArgument,
string messageName){
try{
    //Create the instance of an custom application message.
    iAnywhere.RfidNet.Mobile.ApplicationMessage textMessage =
        new iAnywhere.RfidNet.Mobile.ApplicationMessage();

    //Set the message data.
    textMessage.SetMessageData( messageData);

    //Set up the message argument.
    textMessage.Argument=messageArgument;

    //Setup the message name.
    textMessage.Name = messageName;

    //Send the message to the server.
    mgr.SendApplicationMessage(textMessage);
    return true;
}
catch(Exception e){
    //Catch any failed messages. This can occur if the
    //required dlls are missing, there is a problem with the
```

```
    //network or the server is not configured.  
    return false;  
  }  
}
```

For more information about receiving application messages at the RFID Anywhere server, often from a business module, see [“Application message events from Mobile controller” \[RFID Anywhere Developer's Guide\]](#).

Disconnecting from the RFID Anywhere mobile system

It is important when developing a mobile application with RFID Anywhere to disconnect from the RFID system. By disconnecting from the RFID system the application unregisters all triggers and RFID Anywhere returns them to the default for the device. A code sample is below.

```
/**  
 * Clean up resources used by the RfidNetManager.  
 * */  
  
private void CloseRfidManager(){  
  RfidNetManager mgr = RfidNetManager.GetInstance(false);  
  if (mgr != null){  
    mgr.Dispose();  
  }  
}
```

Index

D

- developing RFID Anywhere applications for mobile devices, 15
- documentation
 - providing feedback, vii
 - RFID Anywhere, vi

F

- feedback
 - documentation, vii
 - providing, vii

R

- RFID Anywhere
 - documentation, vi
- RFID Anywhere Mobile Support Guide, v
