BEA WebLogic RFID Edge Server™

Reader Reference

Version 2.1
Revised: June 29, 2006

Copyright

Restricted Rights Legend

Trademarks and Service Marks

Contents

1. Introduction and Roadmap

2. Configuring RFID Devices

4. Controlling RFID Devices via PLC

5. Triggers

Index

CHAPTER 1

Introduction and Roadmap

The following sections describe the scope and organization of this guide—*WebLogic RFID Reader Reference*.

Document Scope and Audience

This document describes configuration properties for the various RFID devices supported by WebLogic RFID Edge Server. It does not explain other RFID Edge Server functionality such as the ECSpec Editor. Refer to "Related Documentation" on page 1-2 to determine which other documents you may need to consult.

This document is intended for the following audiences:

- System Administrators—IT professionals who are responsible for configuring and deploying RFID devices. Related responsibilities may include installing, testing, maintaining, and upgrading RFID Edge Servers. System administrators understand networking and Web protocols, as well as Java and XML.
- Application Developers—Java programmers who use the Application Level Events (ALE) API to develop client applications and work with other engineering, quality assurance

(QA), and database teams. Application developers have in-depth, working knowledge of Java.

Guide to This Document

This document is organized as follows:

- This chapter, "Introduction and Roadmap," describes the scope of this guide, summarizes new features, and lists related documentation.
- Configuring RFID Devices provides a list of supported RFID devices and describes the configuration properties for each device.
- Configuring and Controlling Stack Lights describes how to configure and control stack lights.
- Controlling RFID Devices via PLC describes how the RFID Edge Server can communicate with and control RFID devices by interacting with a programmable logic controller (PLC).
- Triggers describes external events (triggers) that you use to signal the beginning or end of an event or programming cycle.

Related Documentation

The following WebLogic RFID Edge Server documentation contains information that is relevant to *WebLogic RFID Reader Reference*:

- *Installing WebLogic RFID Edge Server* describes how to install and configure WebLogic RFID Edge Server.
- *Using the RFID Edge Server Administration Console* describes how to use the RFID Administration Console to configure ECSpecs, ECReports, RFID devices, filters, and workflows.
- *Programming with the ALE and ALEPC APIs* describes the Application Level Events (ALE) and Application Level Events Programming Cycle (ALEPC) programming interfaces (APIs) that you use to create applications that interact with WebLogic RFID Edge Server by reading and writing electronic product code (EPC) tags.
- *Using the Reader Simulator* is online help that describes how to use the reader simulator software included with RFID Edge Server. The Reader Simulator minimally simulates a ThingMagic Mercury4 RFID reader.

New Features in This Release

WebLogic RFID Edge Server includes the following new RFID device features:

- Intermec IF5—GPIO support and antenna RF power control. See "Intellitag IF5 Reader" on page 2-36.
- Zebra R110Xi—Gen 2 support.

Configuring RFID Devices

The following sections list all supported RFID devices and describe the configuration properties for each device.

Supported RFID Readers Summary

You need one or more RFID readers to use the WebLogic RFID Edge Server. All the supported makes and models of readers are shown in Table 2-1. The reader models marked with an asterisk (*) have substantially similar command sets and capabilities as others in the same row, but they have not been as extensively tested.

Table 2-1 Supported RFID Readers

Make	Model	Read/Write	Tag Classes Supported
Accraply	ALX-92X series	WO	EPC Class 1
Alien	ALR-9750 (Nanoscanner) 915 MHz RFID	RW	EPC Class 1

Table 2-1 Supported RFID Readers (Continued)

Alien	ALR-9780 915 MHz RFID ALR-8780 866 MHz RFID*	RW	EPC Class 1
Alien	ALR-9780 915 MHz RFID	RO	EPC Class 1 Gen 2
Alien	ALR-9800	RO	EPC Class 0, 1 EPC Class 1 Gen 2
Avery	6405	WO	EPC Class 1
AWID	MPR-2010AN, MPR-2080*	RO	EPC Class 0
AWID	MPR-2010AN, MPR-2080*	RW	EPC Class 0+, 1
AWID	MPR-3014	RW	EPC Class 0+, 1 EPC Class 1 Gen 2
CAEN	A928	RW	ISO 18000-6B EPC EPC Class 1 Gen 2
DataLogic	DS6300-105-010	RO	N/A (bar code reader)
Escort Memory Systems (EMS)	LRP820S, LRP2000	RO	ISO 15693
Identec Solutions	i-CARD 3	RO	ILR-Q and i-D (active tags)
Intermec	Intellitag IF5	RW	Intellitag G1 ISO 18000-6B EPC Class 1 Gen 2
Intermec	PM4i	RW	ISO 18000-6B EPC Class 1 Gen 2
Paxar	Monarch 9855	WO	EPC Class 1
Printronix	T5000e Smart Label	WO	EPC Class 0+, 1
Printronix	T5000r/SL5000r Smart Label	WO	EPC Class 0+, 1 EPC Class 1 Gen 2
SAMSys	MP9320 2.7	RW	EPC Class 1
SAMSys	MP9320 2.7	RO	ISO 18000-6B v1.19 EPC

Table 2-1 Supported RFID Readers (Continued)

SAMSys	MP9320 2.8	RW	EPC Class 1 EPC Class 1 Gen 2
SAMSys	MP9320 2.8	RO	ISO 18000-6B v1.19 EPC
SAMSys	MP 9210	RW	ISO 15693 Philips I-Code
Symbol (Matrics)	RDR-001	RO	EPC Class 0
Symbol (Matrics)	AR400	RO	EPC Class 0, 0+ EPC Class 1 Gen2
		RW	EPC Class 1 Gen1
Symbol (Matrics)	XR400	RO	EPC Class 0, 0+
		RW	EPC Class 1 Gen 1/Gen 2
Symbol (Matrics)	DC400*	RO	EPC Class 0, 0+, EPC Class 1 Gen 1/Gen 2
Tagsys	Medio L100, Medio L200*	RW	ISO 15693 Philips I-Code
ThingMagic Mercury3	Mercury3, Sensormatic Agile 1*	RW	EPC Class 1
ThingMagic Mercury4	Mercury4, Sensormatic Agile 2*, Omron V740*	RO	EPC Class 0, 0+
ThingMagic Mercury4	Mercury4, Sensormatic Agile 2*, Omron V740*	RW	EPC Class 1 EPC Class 1 Gen 2
Zebra	R110XiIIIPlus	WO	EPC Class 0+
Zebra	R110Xi	WO	EPC Class 0+, 1 EPC Class 1 Gen 2
Zebra	R4MPlus	WO	EPC Class 1

Note: Supported read/write types are Read/Write (RW), Read Only (RO), and Write Only (WO).

If you do not have a reader, you can use the reader simulator provided with the RFID Edge Server. Out of the box, it runs on any workstation and simulates a ThingMagic Mercury4 reader; with minor configuration it can simulate a Printronix reader. The reader simulator is useful for software evaluation, application development, and debugging. See *Using the Reader Simulator*.

Configuring Physical Readers

Some readers may require specific configuration prior to use with WebLogic RFID Edge Server. You can configure the RFID Edge Server to communicate with supported readers in one of two ways:

- Edit the reader configuration information using the RFID Devices pane in the Administration Console (the default method for current RFID Edge Server installations). See *Using the RFID Edge Server Administration Console*.
- Edit the *RFID_EDGE_HOME*`/etc/edge.props` file directly to configure these devices, where *RFID_EDGE_HOME* is the directory where you installed the RFID Edge Server software. (This is the default method for RFID Edge Server installations prior to version 1.3.) This file is a Java properties file used to configure the RFID Edge Server. For more information about the `edge.props` file, see *Configuring WebLogic RFID Edge Server*.

  Note: Be aware that these two methods of configuring supported readers are mutually exclusive. All readers are defined either using the Administration Console *or* by editing the `edge.props` file. See Two Approaches to Configuring Readers: edge.props or Administration Console in *Installing WebLogic RFID Edge Server*.

Configuration properties consist of a name (shown in the Property Name column of the tables in this document) and a value (described in the Property Value and Description column). When written out in the `edge.props` file, they appear in the following format:

```
com.connecterra.ale.reader.<deviceID>.<prop1> = <prop1value>
```

If you are using the Administration Console to add and configure readers, you enter the configuration information in reader-specific fields. In this case, no reader configuration information will be written to the `edge.props` file; instead, the configuration information you enter will be saved in the persistence store kept by the RFID Edge Server.

Useful tips for configuring readers are included in reader-specific sections. For troubleshooting information that applies to all readers, see "General Troubleshooting" on page 2-85.

RFID Readers Configuration Properties

The following sections provide configuration information for all the supported makes and models of readers.

Accraply

This section describes driver configuration information for the Accraply label printer. This device is capable of connecting to a reader (referred to as the *Logical Reader Name for RFID Encoding* property value) for tag write operations.

The Accraply reader uses the Easy Plug Label Scripting language. See "Using the Easy Plug Label Scripting Language" on page 2-24.

Table 2-2 lists the Accraply driver configuration properties.

Table 2-2 Accraply Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.AccraplyPhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the printer's LAN adapter.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer's LAN adapter. The default value is 3001.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds).
Logical Reader Name	prnLogicalReaderName	Yes	The logical reader name assigned to the Accraply printer's print head.
N/A	rfPowerLevel	No	See rfAttenuation. Used only for backward compatibility with releases prior to RFTagAware 1.3.

Table 2-2 Accraply Configuration Properties (Continued)

RF Attenuation	rfAttenuation	No	Default antenna power attenuation setting to be passed to the secondary Logical Reader Name. Valid range: 0 (no attenuation, maximum power) to 160 (maximum attenuation, minimum power), in increments of 10 - each increment representing an additional 1 dB of RF attenuation. Default value is 0.
N/A	defaultFormFilename	No	See easyPlugFilename. Used only for backward compatibility with releases prior to RFTagAware 1.3.
Easy Plug Form	easyPlugFilename	No	Specifies the pathname of a file containing Easy Plug commands defining a smart label form. Typically, the PCSpec `readerParameters` attribute specifies Easy Plug-based label design. This property provides a mechanism for specifying a default label design in the event the PCSpec does not contain this parameter. When present, this file name refers to a text file containing the set of Easy Plug commands specifying the smart label layout and dynamic (variable) content. **Note:** While optional, this property must point to a valid Easy Plug label script file if it is present.
Fail Form	failFormFilename	No	The form which will be used instead of the Easy Plug Form if the secondary Logical Reader Name's tag write operation fails.
Enable RFID Encoding	programRFID	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the printer to program EPC data into an embedded RFID tag. The default value is `true`.

Table 2-2 Accraply Configuration Properties (Continued)

Logical Reader Name for RFID Encoding	secondaryLogicalReaderName	Yes	Specifies a (secondary) logical reader to write the RFID tag in the label. This logical reader must be properly specified and configured.
Retries	retries	No	The number of attempts that the secondary Logical Reader Name should make to write the RFID tag in the label.

Alien

The RFID Edge Server uses three Alien reader drivers:

- Alien ALR-9750—Use this driver for the Alien ALR-9750 (Nanoscanner 915 MHz) RFID reader.
- Alien ALR-9780—Use this driver for the Alien ALR-9780 and ALR-8780 RFID readers.
- Alien ALR-9800—Use this driver for Alien ALR-9800 RFID readers.

Alien ALR-9750

This driver is used to interface the WebLogic RFID Edge Server with an Alien Technology ALR-9750 (Nanoscanner 915 MHz) RFID Reader. The reader driver implements the Alien text-based communications protocol for configuring and operating its RFID readers. This text-based reader/ host protocol is specified in the *Nanoscanner Reader User Guide* (Alien Doc # 8101024-000 Rev B).

Table 2-3 lists the Alien ALR-9750 reader driver configuration properties.

Table 2-3 Alien ALR-9750 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	For backward compatibility with releases prior to RFTagAware 1.1.2, set to: `com.connecterra.ale.readertypes.AlienReaderGroup` Otherwise, set to: `com.connecterra.ale.readertypes.AlienALR9750PhysicalReader`

Table 2-3 Alien ALR-9750 Configuration Properties (Continued)

Reader Hostname	hostname	Yes	The DNS name or IP address of the Alien reader.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer's LAN adapter. The default value is 23.
Default Rate	defaultRate	No	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the default rate times the number of logical readers.
Socket Timeout	socketTimeout	Yes	The TCP socket timeout interval (milliseconds). The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default is 15000 milliseconds (15 seconds).
User Name	username	Yes	The user name the Edge Server will use for gaining access to the Alien reader. Must be the same as the user name you configured when setting up the Alien reader. **Note:** The user name is case sensitive and must be entirely lowercase.
Password	password	Yes	The password the Edge Server will use for gaining access to the Alien reader. Must be the same as the password you configured when setting up the Alien reader. **Note:** The password is case sensitive and must be entirely lowercase.
Disable Programming Cycle Check	disableProgrammingCycle Check	No	An optional Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the "check operation" (verification that there is a single tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation.

Table 2-3 Alien ALR-9750 Configuration Properties (Continued)

Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF antenna. At least one logical reader name must be specified or the Edge Server will generate an error on startup.
Antenna 0 Filter Names Antenna 1 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.

The Alien reader obtains its IP network configuration dynamically via DHCP, or statically through one of the reader configuration interfaces. Refer to the Alien *Nanoscanner Reader User Guide* for further details.

The Edge Server Alien reader driver assumes that the Alien reader has also been configured, through the Alien Web interface or command-line interface, with the following configuration settings. These configuration settings must be saved to the reader's flash memory so that reader reboots do not result in their loss.

```
Alien>Set Username=<username>
Alien>Set Password=<password>
```

Alien ALR-9780

This driver is used to interface the WebLogic RFID Edge Server with two models of Alien Technology readers: the ALR-9780 915 MHz RFID reader and the ALR-8780 866 MHz RFID reader.

The reader driver implements the Alien text-based communications protocol for configuring and operating its RFID readers. This text-based reader/host protocol is specified in the *ALR-9780 Reader Interface Guide* (Alien Doc # 8101938-000 Rev 01).

Table 2-4 lists the Alien ALR-9780 reader driver configuration properties.

Table 2-4 Alien ALR-9780 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Set to: `com.connecterra.ale.readertypes.AlienALR9780PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the Alien reader.
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the Alien reader. The default is 23.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the default rate times the number of logical readers.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default is 15000 milliseconds (15 seconds).
User Name	username	Yes	The user name the Edge Server will use for gaining access to the Alien reader. Must be the same as the user name you configured when setting up the Alien reader. **Note:** The user name is case sensitive and must be lowercase.
Password	password	Yes	The password the Edge Server will use for gaining access to the Alien reader. Must be the same as the password you configured when setting up the Alien reader. **Note:** The password is case sensitive and must be lowercase.

Table 2-4 Alien ALR-9780 Configuration Properties (Continued)

Reader Network Timeout	readerNetworkTimeout	No	The amount of time (in seconds) the reader maintains an idle connection with the Edge Server before closing the connection. The integer property value must be greater than 0. Valid range is 1-65535 seconds; default value is 90 seconds.
Read RF Attenuation	readRfAttenuation	No	Sets antenna power attenuation for tag read operations. Valid range: from 0 (no attenuation, maximum power) to 160 (maximum attenuation, minimum power), in increments of 10; each increment representing an additional 1 dB of RF attenuation. Default value is 0.
Write RF Attenuation	writeRfAttenuation	No	Sets antenna power attenuation for program tag operations. Valid range: from 0 (no attenuation, maximum power) to 160 (maximum attenuation, minimum power), in increments of 10; each increment representing an additional 1 dB of RF attenuation. Default value is 0. This property may be overridden at run-time by the application reader parameter `rfAttenuation`.
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	An optional Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the “check operation” (verification that there is a single tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation.

Table 2-4 Alien ALR-9780 Configuration Properties (Continued)

Enable Global Scroll Mode	enableGlobalScrollMode	No	An optional Boolean property specifying whether tags are to be acquired in the Global Scroll mode. The default value is `false` (the driver acquires tags in the Inventory mode).
Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF antenna. At least one logical reader name must be specified or the Edge Server will generate an error on startup. If multiple UHF antennas are assigned the *same* logical reader name, then the driver will treat them as a single combined antenna. The lowest-numbered antenna in the combined antenna grouping is checked to set antenna-specific properties and used for tag programming.
Antenna 0 Acq. Cycles Antenna 1 Acq. Cycles Antenna 2 Acq. Cycles Antenna 3 Acq. Cycles	uhf1AcqCycles uhf2AcqCycles uhf3AcqCycles uhf4AcqCycles	No	The number of acquisition cycles that are performed each time a logical reader conducts a read cycle (is issued a "Get TagList" command). Each logical reader operates with its own value. Integer property values range from 1 to 255. The default value is 1. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.
Antenna 0 Wake Count at Start of Cycle Antenna 1 … Antenna 2 … Antenna 3 …	uhf1AcqEnterWakeCount uhf2AcqEnterWakeCount uhf3AcqEnterWakeCount uhf4AcqEnterWakeCount	No	The number of RF wake commands the Alien reader issues at the start of each acquisition cycle. Each logical reader operates with its own value. Integer property values range from 0 to 255. The default value is 3. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.

Table 2-4 Alien ALR-9780 Configuration Properties (Continued)

Antenna 0 Field Reads per Cycle Antenna 1 Field Reads per Cycle Antenna 2 Field Reads per Cycle Antenna 3 Field Reads per Cycle	uhf1AcqCount uhf2AcqCount uhf3AcqCount uhf4AcqCount	No	The number of field reads (anti-collision searches) that are performed in each acquisition cycle. Each logical reader operates with its own value. Integer property values range from 1 to 255. The default value is 3. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.
Antenna 0 Sleep Commands per Read Antenna 1 ... Antenna 2 ... Antenna 3 ...	uhf1AcqSleepCount uhf2AcqSleepCount uhf3AcqSleepCount uhf4AcqSleepCount	No	The number of RF sleep commands the Alien reader issues after each field read. Each logical reader operates with its own value. Integer property values range from 0 to 255. The default value is 1. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.
Antenna 0 Wake Commands at end of each read Antenna 1 … Antenna 2 … Antenna 3 …	uhf1AcqExitWakeCount uhf2AcqExitWakeCount uhf3AcqExitWakeCount uhf4AcqExitWakeCount	No	The number of RF wake commands the Alien reader issues at the end of each acquisition cycle. Each logical reader operates with its own value. Integer property values range from 0 to 255. The default value imposed by the RFID Edge Server is 1. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.

Table 2-4 Alien ALR-9780 Configuration Properties (Continued)

Antenna 0 Field Inventory Timeout Antenna 1 Field Inventory Timeout Antenna 2 Field Inventory Timeout Antenna 3 Field Inventory Timeout	uhf1GetTagListTimeout uhf2GetTagListTimeout uhf3GetTagListTimeout uhf4GetTagListTimeout	No	The timeouts (in milliseconds) for field inventories (Alien Reader "Get TagList" commands). Each logical reader operates with its own timeout value. This configuration parameter, rather than the `socketTimeout` property, will serve as the socket timeout value when the Edge Server is awaiting a response to a "Get TagList" command. The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default value is the value of the Socket Timeout property (or its default).
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
Stack Light A Logical Reader Name	stackLightALogicalReaderName	No	Specifies a logical reader to control the left stack light. For more information on stack light configuration, see "Configuring and Controlling Stack Lights" on page 3-1.
Red Light IO Mask	stackLightAredLightIOMask	No	Defines GPIO port output lines to control the red light on the stack. The valid range is 0-15; the default value is 0.
Green Light IO Mask	stackLightAgreenLightIOMask	No	Defines GPIO port output lines to control the green part of the stack light. The valid range is 0-15; the default value is 0.

Table 2-4 Alien ALR-9780 Configuration Properties (Continued)

Stack Light B Logical Reader Name	stackLightBLogicalReaderName	No	Specifies a logical reader to control the right stack light. For more information on stack light configuration, see "Configuring and Controlling Stack Lights" on page 3-1.
Red Light IO Mask	stackLightBredLightIOMask	No	Defines GPIO port output lines to control the red light on the stack. The valid range is 0-15; the default value is 0.
Green Light IO Mask	stackLightBgreenLightIOMask	No	Defines GPIO port output lines to control the green part of the stack light. The valid range is 0-15; the default value is 0.

List of Protocols	rfProtocol	No	A blank-separated list of RF protocols. Alien ALR-9780 is a multi-protocol reader. To improve efficiency, the read operation will only attempt to collect tags encoded with the specified protocols. At present, valid values are:
			`All`
			`epcClass1`
			`epcClass1Gen2`
			The default value is `All`.

The Alien reader obtains its IP network configuration dynamically via DHCP, or statically through one of the reader configuration interfaces. Refer to the Alien *Reader Interface Guide* for further details.

The Edge Server Alien reader driver assumes the Alien reader has also been configured, through the Alien Web interface or command-line interface, with the following configuration settings. These configuration settings must be saved to the reader's flash memory so that reader reboots do not result in their loss.

```
Alien>Set Username=<username>
Alien>Set Password=<password>
```

Alien ALR-9800

This driver is used to interface the WebLogic RFID Edge Server with the Alien Technology ALR-9800 reader.

Table 2-5 Alien ALR-9800 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Set to: `com.connecterra.ale.readertypes.AlienALR9800PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the Alien reader.
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the Alien reader. The default is 23.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default is 15000 milliseconds (15 seconds).
Read Timeout	readTimeout	No	Specifies the maximum amount of time (in milliseconds) the reader may spend when servicing a request to read tags in a single antenna's field. The default value is 250 milliseconds.
Write Timeout	writeTimeout	No	Specifies the maximum amount of time (in milliseconds) the reader may spend when servicing a request to program a tag across a single antenna. The default value is 250 milliseconds.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the default rate times the number of logical readers.
User Name	username	Yes	The user name the Edge Server will use for gaining access to the Alien reader. Must be the same as the user name you configured when setting up the Alien reader. **Note:** The user name is case sensitive and must be lowercase.

Table 2-5 Alien ALR-9800 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Password	password	Yes	The password the Edge Server will use for gaining access to the Alien reader. Must be the same as the password you configured when setting up the Alien reader. **Note:** The password is case sensitive and must be lowercase.
Reader Network Timeout	readerNetworkTimeout	No	The amount of time (in seconds) the reader maintains an idle connection with the Edge Server before closing the connection. The integer property value must be greater than 0. Valid range is 1-65535 seconds; default value is 90 seconds.
Read RF Attenuation	readRfAttenuation	No	Sets antenna power attenuation for tag read operations. Valid range: from 0 (no attenuation, maximum power) to 160 (maximum attenuation, minimum power), in increments of 10; each increment representing an additional 1 dB of RF attenuation. Default value is 0.
Write RF Attenuation	writeRfAttenuation	No	Sets antenna power attenuation for program tag operations. Valid range: from 0 (no attenuation, maximum power) to 160 (maximum attenuation, minimum power), in increments of 10; each increment representing an additional 1 dB of RF attenuation. Default value is 0. This property may be overridden at run-time by the application reader parameter `rfAttenuation`.
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	An optional Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the “check operation” (verification that there is a single tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation.

Table 2-5 Alien ALR-9800 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Enable Global Scroll Mode	enableGlobalScrollMode	No	An optional Boolean property specifying whether tags are to be acquired in the Global Scroll mode. The default value is `false` (the driver acquires tags in the Inventory mode).
Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF antenna. At least one logical reader name must be specified or the Edge Server will generate an error on startup. If multiple UHF antennas are assigned the *same* logical reader name, then the driver will treat them as a single combined antenna. The lowest-numbered antenna in the combined antenna grouping is checked to set antenna-specific properties and used for tag programming.
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
Antenna 0 Acq. Cycles Antenna 1 Acq. Cycles Antenna 2 Acq. Cycles Antenna 3 Acq. Cycles	uhf1AcqCycles uhf2AcqCycles uhf3AcqCycles uhf4AcqCycles	No	The number of acquisition cycles that are performed each time a logical reader conducts a read cycle (is issued a “Get TagList” command). Each logical reader operates with its own value. Integer property values range from 1 to 255. The default value is 1. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.

Table 2-5 Alien ALR-9800 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Antenna 0 Wake Count at Start of Cycle Antenna 1 … Antenna 2 … Antenna 3 …	uhf1AcqEnterWakeCount uhf2AcqEnterWakeCount uhf3AcqEnterWakeCount uhf4AcqEnterWakeCount	No	The number of RF wake commands the Alien reader issues at the start of each acquisition cycle. Each logical reader operates with its own value. Integer property values range from 0 to 255. The default value is 3. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.
Antenna 0 Field Reads per Cycle Antenna 1 Field Reads per Cycle Antenna 2 Field Reads per Cycle Antenna 3 Field Reads per Cycle	uhf1AcqCount uhf2AcqCount uhf3AcqCount uhf4AcqCount	No	The number of field reads (anti-collision searches) that are performed in each acquisition cycle. Each logical reader operates with its own value. Integer property values range from 1 to 255. The default value is 3. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.
Antenna 0 Sleep Commands per Read Antenna 1 ... Antenna 2 ... Antenna 3 ...	uhf1AcqSleepCount uhf2AcqSleepCount uhf3AcqSleepCount uhf4AcqSleepCount	No	The number of RF sleep commands the Alien reader issues after each field read. Each logical reader operates with its own value. Integer property values range from 0 to 255. The default value is 1. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.
Antenna 0 Wake Commands at end of each read Antenna 1 … Antenna 2 … Antenna 3 …	uhf1AcqExitWakeCount uhf2AcqExitWakeCount uhf3AcqExitWakeCount uhf4AcqExitWakeCount	No	The number of RF wake commands the Alien reader issues at the end of each acquisition cycle. Each logical reader operates with its own value. Integer property values range from 0 to 255. The default value imposed by the RFID Edge Server is 1. See chapters 3 and 4 of the *Alien Technology Reader Interface Guide* for further information on this Acquire Parameter.

Table 2-5 Alien ALR-9800 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Antenna 0 Field Inventory Timeout Antenna 1 Field Inventory Timeout Antenna 2 Field Inventory Timeout Antenna 3 Field Inventory Timeout	uhf1GetTagListTimeout uhf2GetTagListTimeout uhf3GetTagListTimeout uhf4GetTagListTimeout	No	The timeouts (in milliseconds) for field inventories (Alien reader "Get TagList" commands). Each logical reader operates with its own timeout value. This configuration parameter, rather than the `socketTimeout` property, will serve as the socket timeout value when the Edge Server is awaiting a response to a "Get TagList" command. The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default value is the value of the Socket Timeout property (or its default).
UHF Antenna 0 Starting Q Value UHF Antenna 1 Starting Q Value UHF Antenna 2 Starting Q Value UHF Antenna 3 Starting Q Value	uhf1Gen2Q uhf2Gen2Q uhf3Gen2Q uhf4Gen2Q	No	The starting "Q" value used to tune the performance of the Class 1 Gen 2 air protocol. Integer property values range from 0 to 5. The default value is 3. Small Gen 2 tag populations benefit from a small Q value (0-1), while larger Gen 2 tag populations benefit from a higher Q value (2-5).
Stack Light A Logical Reader Name	stackLightALogicalReaderName	No	Specifies a logical reader to control the left stack light. For more information on stack light configuration, see "Configuring and Controlling Stack Lights" on page 3-1.
Red Light IO Mask	stackLightAredLightIOMask	No	Defines GPIO port output lines to control the red light on the stack. The valid range is 0-15; the default value is 0.
Green Light IO Mask	stackLightAgreenLightIOMask	No	Defines GPIO port output lines to control the green part of the stack light. The valid range is 0-15; the default value is 0.
Stack Light B Logical Reader Name	stackLightBLogicalReaderName	No	Specifies a logical reader to control the right stack light. For more information on stack light configuration, see "Configuring and Controlling Stack Lights" on page 3-1.

Table 2-5 Alien ALR-9800 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Red Light IO Mask	stackLightBredLightIOMask	No	Defines GPIO port output lines to control the red light on the stack. The valid range is 0-15; the default value is 0.
Green Light IO Mask	stackLightBgreenLightIOMask	No	Defines GPIO port output lines to control the green part of the stack light. The valid range is 0-15; the default value is 0.
Invert External Outputs	invertOutputs	No	A Boolean property (allowed values are `true` and `false`) specifying whether to invert external ouputs. When inverted, setting an output high drives the voltage low. The default value is `false`.
RF Modulation Mode	rfModulation	No	Valid values are: • `Standard` (default)—good for general purpose use and meets the EPCglobal Dense Reader Mode spectral mask requirements. • `High Speed`—uses high reader-to-tag data rates for better performance in environments with few readers. • `DRM` (Dense Reader Mode)—provides enhanced filtering for better performance in "noisy" environments. Recommended when many readers are operating in the same area. Meets the EPCglobal Dense Reader Mode spectral mask requirements.

RF Protocol	rfProtocol	No	A blank-separated list of RF protocols. Alien ALR-9800 is a multi-protocol reader. To improve efficiency, the read operation will only attempt to collect tags encoded with the specified protocols. At present, valid values are:
			`All`
			`epcClass1`
			`epcClass1Gen2`
			The default value is `All`.

PLC Transport Name	plcTransport	No	The PLC transport name defined in the `edge.props` file. See "Configuring ALR-9800 GPIO" on page 2-22.

Configuring ALR-9800 GPIO

The Alien ALR-9800 reader has eight external output lines, numbered 0 through 7, and four external input lines, numbered 0 through 3. WebLogic RFID Edge Server lets you configure these GPIO like a PLC device. See "Configuring the Edge Server for PLC Communications" on page 4-1. When configuring data items for the ALR-9800, use `out0`, `out1`, `out2`, ...`out7` for outputs, and `in0`, ...`in3` for inputs.

Listing 2-1 illustrates configuring the Alien ALR-9800 reader for controlling a 5-color stack light connected to the reader's external outputs 1 through 5.

Listing 2-1 Alien ALR-9800 GPIO Configuration

```
#In edge.props file
#Define the PLC Transport 'Reader'
com.connecterra.ale.plc.plcTransport.Alien9800GPIO.metaName = Reader
com.connecterra.ale.plc.plcTransport.Alien9800GPIO.pollInterval = 1000

#Define the message convention 'Simple'
com.connecterra.ale.plc.plcMessageConvention.simple.metaName = Simple
com.connecterra.ale.plc.plcMessageConvention.simple.plcTransport =
Alien9800GPIO

#Define the message convention 'Multiple'
com.connecterra.ale.plc.plcMessageConvention.multiple.metaName = Multiple
com.connecterra.ale.plc.plcMessageConvention.multiple.plcTransport =
Alien9800GPIO

#Define outboundMessage for left stack light
com.connecterra.ale.plc.outboundMessage.mystacklightleft.plcMessageConvention
= multiple
com.connecterra.ale.plc.outboundMessage.mystacklightleft.items = out1 out2 out3
out4 out5

#Define inboundMessage
com.connecterra.ale.plc.inboundMessage.AlienInputs.plcMessageConvention =
simple
com.connecterra.ale.plc.inboundMessage.AlienInputs.receiveItem = in1
com.connecterra.ale.plc.inboundMessage.AlienInputs.matchValues = false

#Define left stacklight. Alternatively, this can be done in the Administration
#Console by defining a PLC Stack Light device.
#See "PLC Stack Light Example" on page 4-4.
com.connecterra.ale.reader.stacklightleft.class =
com.connecterra.ale.readertypes.PLCStackLightPhysicalReader
com.connecterra.ale.reader.stacklightleft.plcMessage = mystacklightleft
com.connecterra.ale.reader.stacklightleft.defaultRate = 0
```

```
com.connecterra.ale.reader.stacklightleft.stackLightLogicalReaderName =
StackLightLeft
```

Avery

This section describes driver configuration information for the Avery 6405 RFID label printer. The printer supports the writing (programming) of Class 1 tags embedded within label stock (“smart labels”) and the printing of those labels.

Table 2-6 lists the Avery 6405 reader driver configuration properties.

Table 2-6 Avery 6405 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.Avery6405PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the printer's LAN adapter.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer’s LAN adapter. The default value is 4000.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds).
N/A	defaultFormFilename	No	See easyPlugFilename. Used only for backward compatibility with releases prior to RFTagAware 1.3.

Table 2-6 Avery 6405 Configuration Properties (Continued)

Easy Plug Form	easyPlugFilename	No	Specifies the pathname of a file containing Easy Plug commands defining a smart label form. Typically, the PCSpec `readerParameters` attribute specifies Easy Plug-based label design. This property provides a mechanism for specifying a default label design in the event the PCSpec does not contain this parameter. When present, the file name refers to a text file containing the set of Easy Plug commands specifying the smart label layout and dynamic (variable) content. **Note:** While optional, this property must point to a valid Easy Plug label script file if it is present.
Enable RFID Encoding	programRFID	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the printer to program EPC data into an embedded RFID tag. The default value is `true`.
Logical Reader Name	uhf1LogicalReaderName	Yes	The logical reader name assigned to the Avery printer's single integrated UHF antenna.
Retries	retries	No	Defines the number of labels to retry in case of a failed RFID operation. The valid range is 1 through 10. The default value is 3.

Using the Easy Plug Label Scripting Language

Easy Plug is a scripting language for specifying the layout and contents of printed labels. Easy Plug is described in detail in the Avery document, *Manual Easy Plug, Release 3.00, 11/2003.* Commands specific to the Avery 6405 are described in the document, *How to RFID with Avery 6405*.

The Avery 6405 and Accraply smart label printer drivers send their printers a separate collection of Easy Plug commands with each "smart label" (a printed label with an embedded RFID tag) to be printed and programmed. Users specify a smart label's Easy Plug commands when defining a PCSpec (see *Programming with the ALE and ALEPC APIs*). A PCSpec `readerParameters` field carries the Easy Plug script as a String object; the corresponding reader parameter name is `easyPlugScript` (or one of `easyPlugScript.accraply` or `easyPlugScript.avery` if your WebLogic RFID Edge Server installation supports more than one device that uses EasyPlug).

As an alternative to specifying Easy Plug within a PCSpec, the Edge Server administrator may specify a default Easy Plug script using the reader driver *Easy Plug Form* property. This property provides a mechanism for specifying Easy Plug commands that serve as a default label design in the event the PCSpec `readerParameters` field does not carry a key/value pair containing an Easy Plug script.

The Easy Plug script, whether defined within the PCSpec or contained in a file identified by the `easyPlugFilename` property, must contain only Easy Plug commands. Here is an example script:

```
#!A1
#IMSR100/152.4//6/89/0V
#ER
#R000.00/000.00
#J100#T20.0
#M1/1
#YN100/0/100///THIS IS A TEST
#J50#T25.0
#YN100/0/58///[EPC_TAG_URI]
#Q1/
#!P1
```

Refer to Avery or Accraply documentation for additional information on Easy Plug.

AWID

This section describes driver configuration information for the AWID MPR-2010AN reader.

For the AWID MPR-3014 reader, the configuration properties are the same as for the AWID MPR-2010AN reader, plus the additional properties listed in Table 2-8.

Table 2-7 lists the AWID MPR-2010AN reader driver configuration properties.

Table 2-7 AWID MPR-2010AN Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.AWID2010ANPhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the AWID MPR-2010AN reader.
Reader Port	port	No	Specifies the TCP port the reader driver will target when establishing connections to the AWID MPR-2010AN reader. The default value is 4000.
Default Rate	defaultRate	Yes	The period (in milliseconds) between polls of a logical reader.
Read Timeout	readTimeout	No	The time interval (in milliseconds) allocated to acquiring Class 1 tags. This interval is a portion of the `defaultRate` described earlier. The default value is 500 milliseconds.
Write Timeout	writeTimeout	No	The time interval (in milliseconds) allocated to programming Class 1 tags. The default value is 500 milliseconds.
Sensitivity	sensitivity	No	The sensitivity setting of the reader channel. The range of valid values is from 0 (minimum sensitivity) to 255 (maximum sensitivity). The default setting is 255.
Socket Timeout	socketTimeout	No	The driver's TCP socket timeout interval, expressed in milliseconds. The integer property must be greater than or equal to 0. A timeout of zero is interpreted as an infinite timeout. The default value is 15000 milliseconds.
RF Power Level	rfPowerLevel	No	The RF power setting of the reader. The range of valid values is from 0 (minimum power) to 255 (maximum power). The default value is 0.

Table 2-7 AWID MPR-2010AN Configuration Properties (Continued)

Disable Programming Cycle Check	disableProgrammingCycleCheck	No	Instructs the reader driver not to perform a program cycle check. Valid values are:
			`true`
			`false`
			The default value is `false`. This property can be specified as `true` only if one of the writable protocols is listed in the `rfProtocols` property, otherwise the driver will generate a configuration error.

Logical Reader Name	uhf1LogicalReaderName	Yes	A logical reader name bound to the AWID MPR-2010AN reader's UHF antenna. This reader has only one UHF antenna; hence, only one logical reader.
Logical Reader Filter Names	uhf1LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.

List of Protocols	rfProtocols	Yes	Valid values are:
			`epcClass0`
			`epcClass1`
			The default value is `epcClass1`.

Table 2-8 AWID MPR-3014 Additional Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name Antenna 4 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	A logical reader name bound to the AWID MPR-3014 reader's UHF antenna. This reader supports up to 4 UHF antennas. At least one logical reader name must be specified.

Tips For AWID MPR-2010AN and MPR-3014 Readers

When setting up AWID MPR-2010AN and MPR-3014 readers, follow the instructions provided in the `MPRSetupGuide.pdf` file (from the original CD).

When connecting to the reader for the first time, you need to set up a local network where the host computer is assigned an IP address in the 192.168.xxx space (the default address is 192.168.1.91).

Note: The reader comes with DHCP enabled by default. Disconnect from any external network where a DHCP server might be active while configuring the reader for the first time.

If you do not know the IP address assigned to the reader, you may restore the factory defaults by pressing and holding a hidden button with a sharp object while rebooting the reader. Release the button when the reader powers up.

MPR-3014

This reader supports up to four UHF antennas. At the present time, the reader cannot reliably detect which antenna ports have antennas attached. It relies on user input to enable and disable appropriate antenna ports. Therefore, when configuring logical readers, it is important that you do not attempt to configure antennas that are not physically attached; it is possible to damage the reader by doing this.

MPR-3014 and Combined Antenna Mode

Unlike some readers, MPR-3014 can only support one antenna group. As always, the group is created by assigning the same logical reader name to several uhf antennas. The antenna group must be contiguous; for example, it must start with uhf1, and cannot skip antennas. If you assign the logical reader name “LR1” to uhf antenna 1 and uhf antenna 2, and then attempt to assign a different name to uhf antenna 3, the driver will interpret this as an attempt to create a second combined antenna group, and will generate an error: "AWID MPR-3014 allows only one combined antenna group".

For more information on antenna groups, see “Combined Antenna Mode” on page 2-64.

CAEN

This section describes driver configuration information for the CAEN Technologies A928 RFID reader. Note that the RFID Edge Server requires the separate installation of the CAEN Java library to enable interoperation with the CAEN A928 RFID reader. Please contact your CAEN supplier for access to the file `CAENRFIDLibrary.jar`. This file should be copied into the `lib` subdirectory of your WebLogic RFID Edge Server installation before starting the software.

Table 2-9 lists the CAEN reader driver properties.

Table 2-9 CAEN Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertype.CaenA928PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the reader.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the `defaultRate` times the number of logical readers.
RF Protocol	rfProtocols	No	The RF protocol currently being used by the CAEN A928. The read operation will only attempt to collect tags encoded with the protocol specified. At least one protocol must be specified. The valid values are `epcClass1Gen2` and `iso18000-6B` (the default).
EPC Byte Offset	epcByteOffset	No	Specifies the byte offset of the EPC within a tag's addressable memory. Values may range from 0 to the amount of available tag memory; the default value is 0.
Byte Length	byteLength	No	Data length in bytes (1-16).
Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF antenna. At least one logical reader name must be specified or the Edge Server will generate an error on startup.

Table 2-9 CAEN Configuration Properties (Continued)

Read RF Power, mW	uhf1ReadRfPower uhf2ReadRfPower uhf3ReadRfPower uhf4ReadRfPower	No	The strength of the RF signal used to read tags, in milliwatts. Valid values are 0-4000; the default value is 500. The maximum value allowable is determined by the reader; see the reader documentation for more information.
Write RF Power, mW	uhf1WriteRfPower uhf2WriteRfPower uhf3WriteRfPower uhf4WriteRfPower	No	The strength of the RF signal used to write tags, in milliwatts. Valid values are 0-4000; the default value is 500. The maximum value allowable is determined by the reader; see the reader documentation for more information.
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
IO Gate 1 IO Gate 2 IO Gate 3 IO Gate 4	ioGateLine1 ioGateLine2 ioGateLine3 ioGateLine4	No	This field should be left blank.
IO Gate Logical Reader Name	ioGateLine1.logicalReaderName ioGateLine2.logicalReaderName ioGateLine3.logicalReaderName ioGateLine4.logicalReaderName	No	Specifies the logical reader controlled by this IO line. The value for this property must match one of the configured Logical Reader Names.
IO Mask	ioGateLine1.ioMask ioGateLine2.ioMask ioGateLine3.ioMask ioGateLine4.ioMask	No	Specifies the IO mask associated with this IO line. Values range from 0 to 15. A value of 0 means "disable IO gating signal".

DataLogic

This section describes driver configuration information for the DataLogic DS6300-105-010 bar code reader. By default, the DataLogic bar code reader reads up to 12 alphanumeric digits (a-z, 0-9).

Table 2-10 lists the DataLogic reader driver configuration properties.

Table 2-10 DataLogic Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.DatalogicPhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the DataLogic reader.
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the DataLogic reader. The default is 23.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the `defaultRate` times the number of logical readers.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default is 15000 milliseconds (15 seconds).
Logical Reader Name	bcrLogicalReaderName	Yes	A logical reader name specifying the DataLogic bar code reader.

Escort Memory Systems (EMS)

This section describes driver configuration information for the Escort Memory Systems (EMS) LRP820S and LRP2000 readers.

Table 2-11 lists the EMS reader driver configuration properties.

Table 2-11 EMS Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.EMSLRPPhysicalReader`
N/A	gatewayName	No	See hostname. Used only for backward compatibility with releases prior to RFTagAware 1.3.
Reader Hostname	hostname	Yes	The DNS name or IP address of the MOXA serial-to-Ethernet adapter to which the reader is connected.
N/A	gatewayPort	No	See port. Used only for backward compatibility with releases prior to RFTagAware 1.3.
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the MOXA serial-to-Ethernet adapter. The default value is 8080.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the default rate times the number of logical readers.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The integer property value must be greater than or equal to 0. A timeout of 0 is interpreted as an infinite timeout. The default is 15000 milliseconds (15 seconds).

Table 2-11 EMS Configuration Properties (Continued)

EPC Byte Offset	epcByteOffset	No	Specifies the byte offset of the EPC within a tag's addressable memory. The readers supported by this driver read and write ISO 15693 tags that have 48 bytes of reprogrammable memory; thus, the integer property values range from 0 to 47. The default value is 0.

EPC Byte Ordering	epcByteOrdering	No	Specifies the byte ordering of the EPC within a tag's addressable memory. Allowed property values are:
			`bigEndian`
			`littleEndian`
			Values are case insensitive.

Reader Model	model	Yes	Specifies the particular model of LRP series reader. Valid values are:
			`LRP820S`
			`LRP2000`

Logical Reader Name	hf1LogicalReaderName	Yes	Specifies the logical reader name for the single HF antenna attached to the EMS LRP reader (the LRP2000 and LRP820S readers each support a single HF antenna).
Logical Reader Filter Names	hf1LogicalReaderFilterNames	No	A blank-separated list of filter names for the logical reader. This value is defined in the Administration Console or in the `edge.props` file.
Command Timeout	hf1CommandTimeout	No	Specifies the command timeout value (in milliseconds) that the EMS LRP reader is provided in the commands the driver issues. Integer property values range from 50 to 65,534 (`0xFFFE`). The default value is 250 milliseconds.

The WebLogic RFID Edge Server requires the use of a MOXA serial-to-Ethernet adapter for LAN access to EMS LRP readers.

Identec Solutions

This section describes driver configuration information for the Identec Solutions i-CARD 3 reader. You must install the manufacturer's software on the machine containing the i-CARD 3 reader.

Table 2-12 lists the Identec Solutions i-CARD 3 reader driver configuration properties.

Table 2-12 Identec Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.IdentecICard3PhysicalReader`
Reader Hostname	hostname	Yes	The hostname or IP address of the host computer where the i-CARD 3 reader has been installed.
Reader Port	port	No	The HTTP port on the host computer targeted by web service clients. The default value is 80.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. If multiple logical readers are active simultaneously, then each logical reader will be read at an interval equal to the defaultRate times the number of logical readers.
Tx Power	txPower	No	Antenna transmission power in dB. The valid values range from -30 to 6. The default value is 6. (Default value not in GUI)
Tag Slots	tagSlots	No	The number of tags the reader expects to respond during a read cycle (for example, setting this field to 32 gives a window of opportunity large enough for exactly 32 tags to respond). In practice, we recommend setting this value to be approximately twice the number of tags you expect to respond. The valid values range from 16 to 4096. The default value is 32. (Default value not in GUI)

Table 2-12 Identec Configuration Properties (Continued)

Enable Rx Boost	enableRxBoost	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader has enabled a feature that allows it to better receive signals from the tags. The reception power is -80dB when this setting is enabled, and -60dB when it is not. The default value is true.
Enable Blink	enableBlink	No	A Boolean (allowed values are true and false) specifying whether the LED on the active tag blinks when it responds to the reader. The default value is `true`.
Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReader Name uhf2LogicalReader Name uhf3LogicalReader Name uhf4LogicalReader Name	Must configure one	The logical reader name for the i-CARD 3 reader (PCMCIA card). Each card installed on the same computer should be assigned a unique logical reader name. At least one logical reader name must be specified. **Note:** Configuring more antennas than there are i-CARD 3 readers will tend to cause errors, and should be avoided.
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReader FilterNames uhf2LogicalReader FilterNames uhf3LogicalReader FilterNames uhf4LogicalReader FilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.

Intermec

The following sections describe driver configuration information for the Intermec Intellitag IF5 Reader and EasyCoder PM4i Printer:

- "Intellitag IF5 Reader" on page 2-36
- "PM4i Printer" on page 2-40

Intellitag IF5 Reader

Table 2-13 lists the Intermec Intellitag IF5 reader driver configuration properties.

Table 2-13 Intermec Intellitag Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.IntermecIF5PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the reader.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the reader. The default value is 2189.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. If multiple logical readers are active simultaneously, then each logical reader will be read at an interval equal to the `defaultRate` times the number of logical readers.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds). This value must be greater than 0.

RF Protocol	rfProtocols	No	The RF protocol currently being used by the Intermec reader. The Intellitag IF5 is a multi-protocol reader. To improve efficiency, the read operation will only attempt to collect tags encoded with the specified protocols. At least one protocol must be specified.
			The valid values are:
			`epcClass1Gen2`
			`iso18000-6B`
			`V119`
			`MIXED` (`iso18000-6B` and `V119`)
			The default value is `MIXED`.

Table 2-13 Intermec Intellitag Configuration Properties (Continued)

ID Tries	idTries	No	Maximum number of identify attempts on a given tag. Values range from 1-255; the default value is 3.
Read Tries	readTries	No	Maximum number of read attempts on a given tag. Values range from 1-255; the default value is 3.
Write Tries	writeTries	No	Maximum number of write attempts on a given tag. Values range from 1-255; the default value is 3.
EPC Byte Offset	epcByteOffset	No	Specifies the byte offset of the EPC within a tag's addressable memory. Values may range from 0 to the amount of available tag memory, minus the amount specified in the *Byte Length* property (below); the default value is 0. When writing ISO18000-6B tags, this value should be equal to or greater than 8. When the Class 1 Gen 2 protocol is selected, this value is ignored by the driver.
Byte Length	byteLength	No	Data length in bytes (0-16). The default value is 8. When the Class 1 Gen 2 protocol is selected, this value is ignored by the driver.
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	An optional Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the "check operation" (verification that there is a single tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation. Note that if the check operation is disabled (this property value is set to `true`) and multiple tags are present in the antenna's field when conducting a tag programming operation, then all tags in the field will be programmed with the same EPC.

Table 2-13 Intermec Intellitag Configuration Properties (Continued)

Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	Specifies the logical reader name for each antenna. At least one logical reader name must be specified. If one of these properties is left undefined, there will be no logical reader associated with the corresponding antenna, and that antenna will not be accessible via the ALE API.
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
Antenna 0 Read RF Power Antenna 1 Read RF Power Antenna 2 Read RF Power Antenna 3 Read RF Power	uhf1ReadRfPower uhf2ReadRfPower uhf3ReadRfPower uhf4ReadRfPower	No	The percentage of maximum RF antenna power for read operations. Valid values are 0-100; the default value is 100.

Table 2-13 Intermec Intellitag Configuration Properties (Continued)

Antenna 0 Write RF Power Antenna 1 Write RF Power Antenna 2 Write RF Power Antenna 3 Write RF Power	uhf1WriteRfPower uhf2WriteRfPower uhf3WriteRfPower uhf4WriteRfPower	No	The percentage of maximum RF antenna power for write operations. Valid values are 0-100; the default value is 100.
PLC Transport Name	plcTransport	No	The PLC transport name defined in the `edge.props` file. See "Configuring Intermec IF5 GPIO" on page 2-39.

Configuring Intermec IF5 GPIO

The Intermec IF5 reader has four external output lines, numbered 1 through 4, and four external input lines, numbered 1 through 4. WebLogic RFID Edge Server lets you configure these GPIO like a PLC device. See "Configuring the Edge Server for PLC Communications" on page 4-1. When configuring data items for the Intermec IF5, use `out1`, `out2`, `...out4` for outputs, and `in1`, `...in4` for inputs.

Listing 2-2 illustrates configuring the Intermec IF5 reader for controlling a 4-color stack light connected to the reader's external outputs 1 through 4. Note that while the reader only has 4 outputs, 5 data items must be defined in the PLC message (one for each color, White, Blue, Green, Amber, Red). Use `NA` for the unused light. In the following example, outputs are defined for White, Blue, Green, and Amber lights; Red is unused.

Listing 2-2 Intermec IF5 GPIO Configuration

```
#In edge.props file
#Define the PLC Transport `Reader'
com.connecterra.ale.plc.plcTransport.IF5GPIO.metaName = Reader
com.connecterra.ale.plc.plcTransport.IF5GPIO.pollInterval = 1000

#Define the message convention `Simple'
com.connecterra.ale.plc.plcMessageConvention.simple.metaName = Simple
com.connecterra.ale.plc.plcMessageConvention.simple.plcTransport = IF5GPIO
```

```
#Define the message convention `Multiple'
com.connecterra.ale.plc.plcMessageConvention.multiple.metaName = Multiple
com.connecterra.ale.plc.plcMessageConvention.multiple.plcTransport = IF5GPIO

#Define outboundMessage for left stack light
com.connecterra.ale.plc.outboundMessage.mystacklightleft.plcMessageConvention
= multiple
com.connecterra.ale.plc.outboundMessage.mystacklightleft.items = out1 out2 out3
out4 NA

#Define inboundMessage
com.connecterra.ale.plc.inboundMessage.IF5INPUTS.plcMessageConvention = simple
com.connecterra.ale.plc.inboundMessage.IF5INPUTS.receiveItem = in1
com.connecterra.ale.plc.inboundMessage.IF5INPUTS.matchValues = false

#Define left stacklight. Alternatively, this can be done in the Administration
#Console by defining a PLC Stack Light device.
#See "PLC Stack Light Example" on page 4-4.
com.connecterra.ale.reader.stacklightleft.class =
com.connecterra.ale.readertypes.PLCStackLightPhysicalReader
com.connecterra.ale.reader.stacklightleft.plcMessage = mystacklightleft
com.connecterra.ale.reader.stacklightleft.defaultRate = 0
com.connecterra.ale.reader.stacklightleft.stackLightLogicalReaderName =
StackLightLeft
```

PM4i Printer

Table 2-14 lists the Intermec PM4i label printer driver configuration properties.

Table 2-14 Intermec PM4i Printer

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.IntermecPM4iPhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the printer.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer. The default value is 9100.

Table 2-14 Intermec PM4i Printer (Continued)

Field Label	Property Name	Required?	Property Value and Description
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds). This value must be greater than 0.
UHF Antenna 1 Logical Reader Name	uhf1LogicalReaderName	Yes	The logical reader name assigned to the device's single integrated UHF antenna.
Enable RFID Encoding	programRFID	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the printer to program EPC data into an embedded RFID tag. The default value is `true`.
Script File	fpFilename	No	Specifies the pathname of a file containing Intermec Fingerprint® commands defining a label form. Typically the Fingerprint script-based label design is specified within the PCSpec `readerParameters` attribute; the reader parameter name is `fpScript`. This property provides a mechanism for specifying a default label design in the event the PCSpec does not contain PM4i reader parameters. When present, the PM4i reader parameter is a string containing the set of Fingerprint commands specifying a label layout.
Retries Count	retries	No	Defines the number of labels to retry in case of a failed RFID operation. The valid range is 1 through 255, the default value is 3.

Paxar

This section describes driver configuration information for the Paxar Monarch 9855 smart label printer. You will need to consult the *Monarch® 9855™ RFID Printer Operator's Handbook* and enable status polling to allow the printer to respond to status requests from the RFID Edge Server.

Table 2-15 lists the Paxar Monarch 9855 reader driver configuration properties.

Table 2-15 Paxar Monarch Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.Monarch9855PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the printer.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer. The printer default value is 9100.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds). This value must be greater than 0.
Logical Reader Name	uhf1LogicalReaderName	Yes	The logical reader name assigned to the device's single integrated UHF antenna.
Enable RFID Encoding	programRFID	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the printer to program EPC data into an embedded RFID tag. The default value is `true`.
MPCL Form	mpclFilename	No	Specifies the pathname of a file containing MPCL commands defining a smart label's form. Typically the MPCL-based label design will be specified within the PCSpec `readerParameters` attribute. This property provides a mechanism for specifying a default label design in the event the PCSpec does not contain Monarch 9855 reader parameters. When present, the Monarch reader parameter is a string containing the set of MPCL commands specifying a smart label's layout.

Printronix

This section describes driver configuration information for the Printronix T5000e and T5000r/SL5000r series of label printers with integrated RFID UHF encoders. Although the Printronix device has a "reader driver," it does not support tag reading. Instead, it supports the writing (programming) of Class 1, Class 0+, and Class 1 Gen 2 (T5000r/SL5000r) tags embedded within label stock ("smart labels") and the printing of those labels.

Table 2-16 lists the Printronix T5000e and T5000r/SL5000r reader configuration properties.

Table 2-16 Printronix Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.PrintronixT5000ePhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the printer's LAN adapter.
Logical Reader Name	uhf1LogicalReaderName	Yes	The logical reader name assigned to the Printronix device's single integrated UHF antenna.
Enable Direct Connection for Label Printing	enablePrint	No	A Boolean (allowed values are `true` and `false`) specifying whether a direct connection to printer for label printing and RFID programming is enabled. The default value is `true`.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer's LAN adapter. The default value is 9100, the Printronix device's factory default.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds).

Table 2-16 Printronix Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Enable RFID Encoding	programRFID	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the printer to program EPC data into an embedded RFID tag. The default value is `true`.
PGL Form	pglFilename	No	Specifies the pathname of a file containing Printronix PGL commands defining a smart label form. Typically, the PCSpec `readerParameters` attribute specifies PGL-based label design; the reader parameter name is `pglScript`. This property provides a mechanism for specifying a default label design in the event the PCSpec does not contain Printronix reader parameters. When present, the file name refers to a text file (`label.pgl`) containing the set of PGL create-mode commands specifying the smart label layout and non-EPC content. See "Using the Printronix Graphics Language (PGL)" on page 2-46.
Enable PXML Port Connection for Printer Status Monitoring	enablePXML	No	An optional Boolean property (allowed values are `true` and `false`), specifying whether the PXML port for printer status monitoring is enabled. The default value is `false`. If it is set to `true`, enter values for the next two properties to properly configure communications via the PXML port.
PXML Port	pxmlPort	No	The port used by the Edge Server to obtain printer status via the printer's Ethernet adapter. The default value is 3007.

Table 2-16 Printronix Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
PXML Message Description File	pxmlMessageFile	No	The path and name of the file that defines Printronix PXML messages, their associated message IDs, severity level, and description. The contents of this file can be edited. This file is based on Printronix's PXML message definition. See the `PXMLMessage.props` example file in the `etc` subdirectory of your Edge Server installation.
Enable GPIO Port Connection for Printer Status Monitoring	enableGPIO	No	An optional Boolean property (allowed values are `true` and `false`), specifying whether the GPIO port for printer status monitoring is enabled. The default value is `false`. If it is set to `true`, enter values for the next two properties to properly configure communications via the GPIO port.
GPIO Port	gpioPort	No	The general-purpose I/O (GPIO) port used by the Edge Server to obtain printer status via the printer's Ethernet adapter. The default value is 3002. **Note:** The GPIO properties are only available for use after installing and configuring the GPIO Accessory Module, available from the manufacturer.

Table 2-16 Printronix Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
GPIO Port Heart Beat Timeout	gpioHeartBeatTimeout	No	The printer regularly sends "heartbeat" messages to the Edge Server to confirm connectivity. (Consult the manufacturer or printer documentation for information on how often these messages are sent.) This value defines how long, in milliseconds, the Edge Server will wait for a message before attempting to re-establish a connection with the printer. The default value is 11000. This property must be set to a value greater than the interval between heartbeat messages.
GPIO Message Content File	gpioMessageFile	No	The path and name of the file that defines Printronix GPIO messages and their associated message IDs. The contents of this file can be edited, but the file should always contain entries for `Heart_Beat_Normal` and `Heart_Beat_Error` messages. This file is based on Printronix's GPIO message definition. See the `GPIOMessage.props` example file in the `etc` subdirectory of your Edge Server installation.

Using the Printronix Graphics Language (PGL)

The Printronix Graphics Language (PGL) is a scripting language for specifying the layout and content of printed labels. PGL is described in detail in the Printronix document, *IGP/PGL Emulation for T5000e series Printers: Printronix Graphics Language Programmer's Reference Manual* (Printronix document 750929-001E).

The Printronix T5000e/T5000r smart label printer drivers send the printer a separate collection of PGL commands with each "smart label" (a printed label with an embedded RFID tag) to be printed and programmed. Users specify a smart label's PGL commands when defining a PCSpec (see *Programming with the ALE and ALEPC APIs*). A PCSpec `readerParameters` field carries the PGL script as a String object; the reader parameter name is `pglScript`.

As an alternative to specifying PGL within a PCSpec, the Edge Server administrator may specify a default PGL script using the Printronix reader driver *PGL Form* property. This property

provides a mechanism for specifying PGL commands that serve as a default label design in the event the PCSpec `readerParameters` field does not carry a key/value pair containing a PGL script.

The PGL script, whether defined within the PCSpec or contained in a file identified by the *PGL Form* property, must only contain IGP/PGL Create Form mode commands. Below is an example PGL script:

```
SCALE;DOT;200;200
FONT;FACE 92250
ALPHA
POINT;60;50;8;8;*SHIP FROM:*
POINT;90;50;10;10;*ACME*
POINT;120;50;10;10;*Corporation*
POINT;150;50;10;10;*P.O. Box 1000*
POINT;180;50;10;10;*Dallas, TX75261*


POINT;60;365;8;8;*SHIP TO:*
POINT;90;365;10;10;*Retailer Distribution Center*
POINT;120;365;10;10;*200 Main Street*
POINT;150;365;10;10;*Springfield, MA01103*
STOP


HORZ
4;200;40;790
STOP


VERT
3;343;60;200
STOP


ALPHA
POINT;235;50;8;8;*(420)SHIP TO POSTAL CODE:*
POINT;285;100;12;12;*(420) 01103*
POINT;235;385;8;8;*CARRIER:*
POINT;285;400;12;12;*Acme Freightways*
```

```
POINT;330;400;10;10;*PRO: 1234*
POINT;370;400;10;10;*B/L: 5678*
STOP

BARCODE
C128C;XRD3:3:6:6:9:9:12:12;H7;283;75
*01103*
STOP

HORZ
4;425;40;790
STOP

VERT
3;374;200;425
STOP

ALPHA
POINT;465;50;12;12;*EPC:*
AF512;25;POINT;550;70;14;14
STOP

HORZ
4;625;40;779
STOP

ALPHA
POINT;665;50;12;12;*SKU:*
POINT;705;70;14;14;*ABC21270*
STOP
```

```
HORZ
4;750;40;779
STOP


ALPHA
POINT;790;50;12;12;*GTIN:*
POINT;840;270;12;12;*(01) 10036000212706*
STOP


BARCODE
C128C;XRD3:3:6:6:9:9:12:12;H7;850;250
*10036000212706*
STOP
```

The user-defined PGL script must not contain an `RFWRITE` Create Form mode command; the reader driver will insert the necessary `RFWRITE` command if it is called on to program a smart label's embedded RFID tag.

If the user wishes to print a specified EPC value on the smart label, the user-supplied PGL must use special substitution strings, indicating in what representation the tag values are to be printed. The RFID Edge Server recognizes the following substitution string values:

`[EPC_HEX]`	Will be replaced by the hexadecimal representation of the tag value.
`[EPC_ID_URI]`	Will be replaced by the pure identity URN representation of the tag value.
`[EPC_TAG_URI]`	Will be replaced by the tag URN representation of the tag value.

Example:

```
ALPHA
POINT;465;50;12;12;*EPC:*
POINT;550;70;14;14;*[EPC_ID_URI]*
STOP
```

Please refer to Printronix documentation for additional information on IGP/PGL.

Using the Printronix Device Management Language (PXML)

The Printronix Device Management Language (PXML) is a printer device management protocol to control and monitor job and printer status. To use PXML, copy the following required JAR files from Printronix into the `lib` subdirectory of your WebLogic RFID Edge Server installation: `xml.jar`, `commonlib.jar`, and `jdom.jar`.

PXML is an XML-based language that allows a client application to issue commands to a Printronix T5000r/SL5000r printer and receive responses from the printer, including:

- Printer power up modes
- Printer status reporting
- Job status reporting, including RFID tag and online data validation information

Responses can be either *solicited* (in response to a command) or *autonomous* (generated by events that occur during printer operation).

PNE is the Printronix proprietary device management protocol used by the Printronix PrintNet Enterprise software. PXML is the XML-based device management protocol to interface with third party software. Currently there is no auto switching between the two protocols and only one device management protocol can be active at any time. The printer factory default is PNE mode. Therefore, third party software must configure the printer for PXML before it can be used. This can be done either by sending the `PTX_SETUP` commands to the printer or setting the printer front panel menu.

Consult the Printronix documentation for detailed PXML setup information.

SAMSys

The RFID Edge Server uses two SAMSys reader drivers:

- SAMSys MP9320—Use this driver for the SAMSys MP9230 reader versions 2.7 and 2.8.
- SAMSys MP9210—Use this driver for the SAMSys MP9210 reader.

SAMSys MP9320

This section describes driver configuration information for the SAMSys MP9320 RFID reader. SAMSys MP9320 2.8 readers feature a single-color stack light, which requires no configuration.

The SAMSys MP9320 reader drivers use the following properties

Table 2-17 SAMSys MP9320 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.SAMSysMP9320PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the reader.
Reader Port	port	No	The TCP port the SAMSys MP9320 reader driver will target when establishing connections to the SAMSys reader. The default value is 2101.
Default Rate	defaultRate	Yes	The period (in milliseconds) between polls of a logical reader. The same value applies to all logical readers provisioned on a single physical reader.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). Must be greater than or equal to zero. A timeout of zero is interpreted as an infinite timeout. The default value is 15000 milliseconds (15 seconds).
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	A Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the Program Cycle Check (verification that there is exactly one programmable tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation. Note that if the check operation is disabled (this property value is set to `true`) and multiple tags are present in the antenna's field when conducting a tag programming operation, then all tags in the field will be programmed with the same EPC.

Table 2-17 SAMSys MP9320 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description

List of Protocols	rfProtocols	No	A blank-separated list of RF protocols. SAMSys MP9320 is a multi-protocol reader. To improve efficiency, the read operation will only attempt to collect tags encoded with the specified protocols. At present, valid values are:
			`epcClass1` (includes `epcClass1-64` and `epcClass1-96`)
			`epcClass1-64` (EPC Class 1 Gen 1 64-bit)
			`epcClass1-96` (EPC Class 1 Gen 1 96-bit)
			`epcClass1Gen2`
			`epcClass1G1G2` (includes `epcClass1` and `epcClass1Gen2`.)
			Note: This choice is Read Only. For tag programming, select a single protocol type.
			`iso18000-6B-64`
			`iso18000-6B-96`
			The default value is `epcClass1`.
			Note: Only one of the ISO18000-6B protocols at a time can be specified. The Edge Server will generate an error if both protocols are listed.

Enable Reader Beep	enableReaderBeep	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the reader to beep when reading an RFID tag. The default value is `false`.

Table 2-17 SAMSys MP9320 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description

Reader Operation Mode	operationMode	No	Indicates the behavior of the SAMSys reader in response to a request for data from an application. A value of `poll` means that the reader is idle between requests, but performs a tag read when it receives a request for data. A value of `autopoll` means that the reader continuously reads tags and stores the data in memory between requests, and forwards all stored tag information when it receives a request from the application.
			Valid values are:
			`poll`
			`autopoll`
			The default value is `poll`.

Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name Antenna 4 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	A set of properties specifying a logical reader name bound to each of the SAMSys MP9320 reader's UHF antennas. At least one logical reader name must be specified. The SAMSys MP9320 reader supports up to four logical readers. If a property within this set is left undefined, there will be no logical reader associated with the corresponding antenna.

Table 2-17 SAMSys MP9320 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names Antenna 4 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
Antenna 1 RF Power Level Antenna 2 RF Power Level Antenna 3 RF Power Level Antenna 4 RF Power Level	uhf1PowerLevel uhf2PowerLevel uhf3PowerLevel uhf4PowerLevel	No	The RF power setting of the reader. The range of valid values is from RF power level 2 (34.7dBm (2.95W)) to RF power level 80 (16 dBm (40 mW)). The default value is RF power level 24 (1W). **Note:** The lower the RF power level value the higher the power output. Refer to the RF power level table in the SAMSys MP9320, *Low Power UHF Antenna Application Note*, Version 2.0.

Setting the Baud Rate

To attain the maximum output from the SAMSys MP9320 2.7 and 2.8 readers, BEA Systems recommends setting the reader to 57600 bps. If you change the baud rate of the reader in the RFID Edge Server, you will need to perform these steps again to keep the baud rate settings the same throughout the hardware and software.

Use the following instructions set the baud rate:

1. Use a serial cable to connect the reader RS232 port to your computer serial port.
2. Run the SAMSys reader RF Command Suite software.
3. Select the **Reader Connection** | **Connect via Serial Port** menu option.

   The software should detect the reader connected to your computer serial port.
4. If you cannot establish a connection, select the menu item **Reader Connection** | **Serial Port Settings**, and use the *Maximum Speed* field to set the correct baud rate to connect to the reader.

5. Open a Web browser and navigate to `http://<IP.address.of.reader>/`.

   This URL will launch a Java applet. You will need to enable Java in your Web browser to see the applet. If you would prefer to perform this configuration via Telnet, skip step 5 and follow the instructions in Appendix C of the *MP9320 2.7 User's Guide*.

6. On the applet that displays, click **Serial Ports** on the left side of the page, change the Baud Rate to `57600`, then click **Save**.

7. In the RF Command Suite software, click the Command tab, and type the command `}Cw,d:scw,b:00161616,f:1!` to set the reader baud rate to 57600.

8. Close the RF Command Suite and the Web browser window.

9. Power-cycle the reader.

   Afterward, the reader external RS232 port and the external serial port on the Digi Connect ME Module are set to a baud rate of 57600.

   Note: If the reader stops responding, use HyperTerminal to connect to the reader, power-cycle it, and press the Enter key three times. This will set the reader to the factory default.

SAMSys MP9210

The SAMSys MP9210 reader driver requires the use of a MOXA serial-to-Ethernet adapter, available from Symbol Technologies.

Table 2-18 lists the SAMSys MP9210 reader configuration properties.

Table 2-18 SAMSys MP9210 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.SAMSysMP9210PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the MOXA serial-to-Ethernet adapter to which the reader is connected.

Table 2-18 SAMSys MP9210 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the MOXA serial-to-Ethernet adapter. The default value is 4001.
Default Rate	defaultRate	Yes	The period (in milliseconds) between polls of a logical reader. The same value applies to all logical readers provisioned on a single physical reader.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). Must be greater than or equal to zero. A timeout of zero is interpreted as an infinite timeout. The default value is 15000 milliseconds (15 seconds).

List of Protocols	rfProtocols	No	A blank-separated list of RF protocols. SAMSys MP9210 is a multi-protocol reader. To improve efficiency, the read operation will only attempt to collect tags encoded with the specified protocols. The valid values are:
			`iso15693`
			`Philips_I_Code`
			The default value is `iso15693`.

Antenna 1 Logical Reader Name	hf1LogicalReaderName	Yes	A set of properties specifying a logical reader name bound to the SAMSys MP9210 reader's HF antenna.
Antenna 1 Filter Names	uhf1LogicalReaderFilterNames	No	A blank-separated list of filter names for the logical reader. This value is defined in the Administration Console or in the `edge.props` file.

Table 2-18 SAMSys MP9210 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Memory Block Offset	memBlockOffset	Yes	The memory block location where EPC data written to the tag begins. For example, an offset of 3 means that the first two memory blocks in the user memory area are skipped and the EPC data begins at the third memory block.
Number of Memory Blocks	memBlockLength	No	The number of memory blocks used in storing EPC data. For example, a 96-bit tag uses three 32-bit blocks.

User Data Byte Ordering	memByteOrder	No	Specifies the byte ordering of the EPC within a tag's addressable memory. Allowed values are:
			`bigEndian`
			`littleEndian`
			Values are case insensitive.

Configuring The MOXA Device

The SAMSys MP9210 reader will be connected to an IP network with a MOXA serial server. The serial configuration is shown below. If the reader's serial configuration is changed, adjust these settings accordingly.

Baud rate:	9600
Parity:	None
Data bits:	8
Stop bit:	1
Flow control:	None
UART FIFO:	Disable
Interface:	RS-232

A null modem is required if using a straight serial cable (DB9, female-to-male).

Symbol (Matrics)

The RFID Edge Server uses two Symbol (Matrics) reader drivers:

- Symbol (Matrics) AR400/XR400—Use this driver for the Symbol (Matrics) AR400 and XR400 readers.
- Symbol (Matrics) RDR-001—Use this driver for the Symbol (Matrics) RDR-001 reader.

Symbol (Matrics) AR400/XR400

Table 2-19 lists the Symbol (Matrics) AR400 and XR400 readers configuration properties.

Table 2-19 Symbol (Matrics) AR400/XR400 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.MatricsAR400PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the reader.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the reader. The default value is 3000.
Default Rate	defaultRate	Yes	The period (in milliseconds) between polls of a logical reader. The same value applies to all logical readers provisioned on a single physical reader.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds).
Frequency Channel	frequencyChannel	No	The frequency channel the AR400 reader will use to read tags. This property MUST ONLY be used with Symbol/Matrics FCC part 90 readers. Integer property values range from 0 to 13, corresponding to 914.25 MHz to 917.25 MHz, in 500 kHz steps. There is no default value. When not specified, the Set Frequency Channel command is not issued.

Table 2-19 Symbol (Matrics) AR400/XR400 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	A Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the Program Cycle Check (verification that there is exactly one programmable tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation. Note that if the check operation is disabled (this property value is set to `true`) and multiple tags are present in the antenna's field when conducting a tag programming operation, then all tags in the field will be programmed with the same EPC.
Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	A set of properties specifying a logical reader name bound to each of the reader's UHF antennas. At least one logical reader name must be specified. (The AR400/XR400 readers support up to four logical readers.) If a property within this set is left undefined, there will be no logical reader associated with the corresponding antenna. **Note:** The AR 400 antennas work in pairs (require two antennas to make a read point, one to receive signals and one to transmit them). If multiple UHF antennas are assigned the same logical reader name, then the driver will treat them as a single combined antenna. The lowest-numbered antenna in the combined antenna grouping is checked to set antenna-specific properties and used for tag programming. If running in this combined antenna mode, see "Combined Antenna Mode" on page 2-64.

Table 2-19 Symbol (Matrics) AR400/XR400 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Antenna 0 Environment Variable Antenna 1 Environment Variable Antenna 2 Environment Variable Antenna 3 Environment Variable	uhf1EnvironmentVariable uhf2EnvironmentVariable uhf3EnvironmentVariable uhf4EnvironmentVariable	No	A property controlling the length of time the AR400 reader gives itself when reading tags within a single antenna's field in response to a single “Read Full Field” Command. Integer property values range from 0 to 8, with full field read times lengthening with increasing environment variable property value. The default environment variable value is 3.
Antenna 0 Power Level Antenna 1 Power Level Antenna 2 Power Level Antenna 3 Power Level	uhf1PowerLevel uhf2PowerLevel uhf3PowerLevel uhf4PowerLevel	No	A set of properties specifying UHF antenna (n=1..4) power levels. Integer property values range from 1 to 255. Power level is logarithmic; hence, 192 is about 50% of and 128 is about 25% of an antenna's maximum power. The default power level is 192.
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
Left Stack Light Logical Reader Name	leftStackLightLogicalReaderName	No	Specifies a logical reader to control the left stack light. See “Configuring and Controlling Stack Lights” on page 3-1 for more information on stack light configuration.

Table 2-19 Symbol (Matrics) AR400/XR400 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Right Stack Light Logical Reader Name	rightStackLightLogicalReaderName	No	Specifies a logical reader to control the right stack light. See "Configuring and Controlling Stack Lights" on page 3-1 for more information on stack light configuration.
List of Protocols	rfProtocol	No	A blank-separated list of RF protocols. The valid values are: `epcClass0` (EPC Class 0) `epcClass1` (EPC Class 1 Gen 1) `epcClass1Gen2` (EPC Class 1 Gen 2) `epcClass1G1G2` (includes `epcClass1` and `epcClass1Gen2`) `all` The default value is `all`.

Stack Light Control Protocol	stacklightProtocol	No	Specifies the protocol to use for stack light control. The default value is `Bytestream`; valid values are:
			`Bytestream`
			`http`
			Note: If the firmware does not work well with the `Bytestream` protocol, try the `http` protocol.

Enable Reader GPIO Mode	enableGPIO	No	An optional Boolean property (allowed values are `true` and `false`), specifying whether GPIO mode is enabled. The default value is `false`. See "Enabling GPIO Mode (Symbol XR400 Reader Only)" on page 2-62.

Table 2-19 Symbol (Matrics) AR400/XR400 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
GPIO Transport Name	gpioTransportName	No	Applies only to the XR400 reader. The PLC transport name defined in the `edge.props` file.
GPIO Port Polling Interval	gpioPollInterval	No	Applies only to the XR400 reader. The period (in milliseconds) between the start of one poll cycle and the start of the next. The default value is 500 milliseconds.

Enabling GPIO Mode (Symbol XR400 Reader Only)

To capture the reader GPIO input, you must enable GPIO mode by performing the following steps:

1. In the Administration Console or the `edge.props` file, set **Enable Reader GPIO Mode** (`enableGPIO`) to `true`.
2. Configure GPIO mode in the reader:
   a. Open a Web browser and enter the following URL: `http://`*`reader_ipaddress`*
   b. Log in to the Symbol XR400 Reader Administrator Console.

      See the Symbol Reader Manual, *XR Series RFID Reader Integrator Guide* (Part Number 72E-71773-03 Rev. A, February 2006), for login instructions.
   c. Under Options on the left, select **Configuration**.
   d. On the Reader Configuration page, select **Advanced Reader (ipAddr:...)**.
   e. On the Modify a Reader Configuration page, select the check box **Do not detect Symbol Light Indication Box** and click **Modify Reader**.
   f. Under Options on the left, select **Commit/Revert**, then click the **Commit** button.
   g. Under Options, select **Logout**.
3. Restart the reader (power off and on).

Configuring the Symbol XR400 GPIO

The Symbol XR400 reader has five external output lines. Listing 2-3 illustrates configuring the Symbol XR400 reader for controlling a stack light connected to the reader's external outputs.

Listing 2-3 Symbol XR400 GPIO Configuration

```
#######################################################
# Define the Reader's GPIO output for stack light use
#######################################################

#In edge.props
#Define the PLC Transport 'Reader'
com.connecterra.ale.plc.plcTransport.myreadergpio.metaName = Reader
com.connecterra.ale.plc.plcTransport.myreadergpio.pollInterval = 500

#Define the message convention 'Multiple'
com.connecterra.ale.plc.plcMessageConvention.mymultiplemsg.metaName = Multiple
com.connecterra.ale.plc.plcMessageConvention.mymultiplemsg.plcTransport =
myreadergpio

#Define outboundMessage message for left stack light
com.connecterra.ale.plc.outboundMessage.mystacklightleft.plcMessageConvention
= mymultiplemsg

#(NA-not applicable, LG-left green, LY-left yellow, LR-left red)
com.connecterra.ale.plc.outboundMessage.mystacklightleft.items = NA NA LG LY LR

#Define left stacklight. Alternatively, this can be done in the Administration
#Console by defining a PLC Stack Light device.
#See "PLC Stack Light Example" on page 4-4.
com.connecterra.ale.reader.stacklightleft.class =
com.connecterra.ale.readertypes.PLCStackLightPhysicalReader
com.connecterra.ale.reader.stacklightleft.plcMessage = mystacklightleft
com.connecterra.ale.reader.stacklightleft.defaultRate = 0
com.connecterra.ale.reader.stacklightleft.stackLightLogicalReaderName =
StackLightLeft

#Define outboundMessage message for right stack light
com.connecterra.ale.plc.outboundMessage.mystacklightright.plcMessageConvention
= mymultiplemsg

#(NA-not applicable, RG-right green, RY-right yellow, RR-right red)
com.connecterra.ale.plc.outboundMessage.mystacklightright.items = NA NA RG RY RR

#Define right stacklight. Alternatively, this can be done in the Administration
#Console by defining a PLC Stack Light device.
#See "PLC Stack Light Example" on page 4-4.
```

```
com.connecterra.ale.reader.stacklightRight.class =
com.connecterra.ale.readertypes.PLCStackLightPhysicalReader
com.connecterra.ale.reader.stacklightRight.plcMessage = mystacklightright
com.connecterra.ale.reader.stacklightRight.defaultRate = 0
com.connecterra.ale.reader.stacklightRight.stackLightLogicalReaderName =
StackLightRight

#define the input trigger
com.connecterra.ale.triggerDriver.plcmessage.class=com.connecterra.ale.trigger
types.PLCTriggerDriver

############################################################
# Define the GPIO input for sensors/photoelectric eyes
############################################################

#In edge.props
#Define the message convention
com.connecterra.ale.plc.plcMessageConvention.mysimplemsg.metaName = Simple
com.connecterra.ale.plc.plcMessageConvention.mysimplemsg.plcTransport =
myreadergpio

#Define inboundMessage message
com.connecterra.ale.plc.inboundMessage.photoeye1.plcMessageConvention =
mysimplemsg

#Input pin 5 for photo eye1
com.connecterra.ale.plc.inboundMessage.photoeye1.receiveItem = input5
com.connecterra.ale.plc.inboundMessage.photoeye1.matchValues = true
com.connecterra.ale.plc.inboundMessage.photoeye2.plcMessageConvention =
mysimplemsg

#Input pin 4 for photo eye2
com.connecterra.ale.plc.inboundMessage.photoeye2.receiveItem = input4
com.connecterra.ale.plc.inboundMessage.photoeye2.matchValues = true

#Define the direction trigger
com.connecterra.ale.triggerDriver.door1.class=com.connecterra.ale.triggertypes
.DirectionTriggerDriver
com.connecterra.ale.triggerDriver.door1.timeout=10000
com.connecterra.ale.triggerDriver.door1.aTrigger=plcmessage:photoeye1
com.connecterra.ale.triggerDriver.door1.bTrigger=plcmessage:photoeye2
```

Combined Antenna Mode

The AR400 and XR400 reader drivers in the RFID Edge Server run a bytestream protocol. The bytestream protocol driver now supports combined antenna operations. To enable combined antenna operations, assign the same logical reader name to multiple physical antenna ports. All

antennas combined within a single logical group must operate with the same set of antenna-specific configuration options (*Power Level*, *Environment Variable*). The driver will use the configuration properties assigned to the lowest numbered antenna port in a group for all antennas in that group. So, if two logical readers are combined into the same group, then the power level and environment variable properties specified for the first will also apply to the second.

Tips For Symbol AR/XR-400 Readers

Before using a Symbol reader with the RFID Edge Server, verify that the reader is not running in "polled" mode. An easy way to do this is to wave a tag in front of the antenna and watch the yellow LED. If it blinks when the tag is in the antenna field, the reader is in "polled" mode. Follow the steps in `Symbol AR400 hardware configuration.doc` to disable polling.

Symbol (Matrics) RDR-001

The Symbol (Matrics) RDR-001 reader driver requires the use of a MOXA serial-to-Ethernet adapter, available from Symbol Technologies.

Table 2-20 lists the Symbol (Matrics) RDR-001 reader configuration properties.

Table 2-20 Symbol (Matrics) RDR-001 Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.MatricsRDR001PhysicalReader`
N/A	gatewayName	No	See hostname. Used only for backward compatibility with releases prior to RFTagAware 1.3.
Reader Hostname	hostname	Yes	The DNS name or IP address of the MOXA serial-to-Ethernet adapter to which the reader is connected.
N/A	gatewayPort	No	See port. Used only for backward compatibility with releases prior to RFTagAware 1.3.

Table 2-20 Symbol (Matrics) RDR-001 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the MOXA serial-to-Ethernet adapter. The default value is 8080.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the `defaultRate` times the number of logical readers.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default is 15000 milliseconds (15 seconds).
Frequency Channel	frequencyChannel	No	The frequency channel the RDR-001 reader will use to read tags. This property MUST ONLY be used with RDR-001 readers running FCC Part 90 Firmware. Integer property values range from 0 to 13, corresponding to 914.25 MHz to 917.25 MHz, in 500 kHz steps. There is no default value.
Node Address	nodeAddress	No	The reader's RS-485 protocol node address. Integer property values range from 0 (00 hex) to 31 (1F hex). The default value is 4.
Antenna 0 Logical Reader Name Antenna 1 Logical Reader Name Antenna 2 Logical Reader Name Antenna 3 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName uhf4LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF antenna. At least one logical reader name must be specified or the Edge Server will generate an error on startup.

Table 2-20 Symbol (Matrics) RDR-001 Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Antenna 0 Environment Variable Antenna 1 Environment Variable Antenna 2 Environment Variable Antenna 3 Environment Variable	uhf1EnvironmentVariable uhf2EnvironmentVariable uhf3EnvironmentVariable uhf4EnvironmentVariable	No	Controls the length of time the RDR-001 reader gives itself when reading tags within a single antenna's field. The value of this property must be an integer in the range 0 to 8, where greater numbers indicate longer time. The default is 3.
Antenna 0 Power Level Antenna 1 Power Level Antenna 2 Power Level Antenna 3 Power Level	uhf1PowerLevel uhf2PowerLevel uhf3PowerLevel uhf4PowerLevel	No	The power level for each UHF antenna. The value of this property must be an integer in the range 1 to 255. Power level is logarithmic; hence, 192 is about 50% of and 128 is about 25% of an antenna's maximum power. The default power level is 192.
Antenna 0 Filter Names Antenna 1 Filter Names Antenna 2 Filter Names Antenna 3 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames uhf4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.

Tagsys

This section describes driver configuration information for the Tagsys Medio L100 and L200 RFID readers. Note that the RFID Edge Server requires the separate installation of the Tagsys Java library to enable interoperation with the Tagsys RFID readers. Please contact your Tagsys supplier for access to the file `com.tagsys.jar`. This file should be copied into the `lib` subdirectory of your WebLogic RFID Edge Server installation before starting the software.

The Tagsys Medio L100 and L200 reader drivers require the use of a MOXA serial-to-Ethernet adapter, available from Symbol Technologies.

Table 2-21 lists the Tagsys Medio L100 and L200 reader configuration properties.

Table 2-21 Tagsys Reader Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.TagsysLX00PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the MOXA serial-to-Ethernet adapter to which the reader is connected.
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the MOXA serial-to-Ethernet adapter. The default value is 4001.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 10000 milliseconds (10 seconds).
Default Rate	defaultRate	Yes	The period (in milliseconds) between polls of a logical reader. The same value applies to all logical readers provisioned on a single physical reader.

RF Protocols	rfProtocols	No	A blank-separated list of RF protocols. The Tagsys readers are multi-protocol readers. To improve efficiency, the read operation will only attempt to collect tags encoded with the specified protocols. At present, valid values are:
			`Philips_I_code`
			`iso15693`
			The default value is `iso15693`.

Table 2-21 Tagsys Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description

Reader Operation Mode	operationMode	No	Indicates the behavior of the Tagsys reader in response to a request for data from an application. A value of `poll` means that the reader is idle between requests, but performs a tag read when it receives a request for data. A value of `autopoll` means that the reader continuously reads tags and stores the data in memory between requests, and forwards all stored tag information when it receives a request from the application.
			Valid values are:
			`poll`
			`autopoll`
			The default value is `poll`.
			To read multiple tags using multiple antennas, `autopoll` mode is recommended.

Enable Slow Communication Mode?	enableSlowComMode	No	An optional Boolean property (allowed values are `true` and `false`). Slow communication mode enforces a data transmission integrity check operation (cyclic redundancy check (CRC)). The default value is `false`. Some types of serial-to-Ethernet adapters might require using slow communication mode.
Read UID/EPC	readEPCUID	No	Specifies whether to read the `UID` (as raw 64-bit data) or the `EPC`. The default is `UID`.
Memory Block Offset	memBlockOffset	Yes	The memory block location where EPC data written to the tag begins. For example, an offset of 3 means that the first two memory blocks in the user memory area are skipped and the EPC data begins at the third memory block.

Table 2-21 Tagsys Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Number of Memory Blocks	memBlockLength	No	The number of memory blocks used in storing EPC data. For example, a 96-bit tag uses three 32-bit blocks. The default is 3.

Enable Little Endian Byte Order?	memByteOrder	Yes	Specifies the byte ordering of the EPC within a tag's addressable memory. Allowed values are:
			`true`
			`false`
			The default value is `false`.

RF Power Level	readRFPower	No	The RF power setting of the Tagsys reader. The range of valid values is from 250 (minimum power) to 4000 (maximum power). The default value is 1000mW.
RF Read Timeout	rfReadTimeout	No	Specifies the amount of RF time (in milliseconds) the reader may spend when servicing a request to read tags in a single antenna's field. The default value is 100 ms.
Expected Tag Count	tagCount	No	Specifies the number of tags expected to be read. The default value is 10.
Time Slots Number	timeslots	No	Specifies the number of time slots used by the RF protocol for anti-collision during tag reading. Valid values are: `1`, `4`, `8`, `16`, `32`, `64`, `128`, `256`. For one tag, use a value of `1`. For multiple tags, try different values to achieve the best performance. If not specified, the driver calculates a value (not necessarily the optimal value).

Table 2-21 Tagsys Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Buffer Depth	bufferDepth	No	Sets the reader's temporary memory storage of tag data in number of tags. For one tag, use a value of `1`. For multiple tags, use a value of `48`. If not specified, the driver calculates a value (not necessarily the optimal value).
RPU Filter Bandwidth	receiverFilter	No	Sets the receiver filtering bandwidth of the Radio Processing Unit (RPU). Valid values are: `BandPass` (between 5 and 600 kHz) `HighPass` (between 80 and 600 kHz) The default is `BandPass`.
Antenna Multiplexing - RF Duration Per Antenna Scan	muxDurationPerScan	No	Applies only to combined antenna operations. Specifies the amount of time (in milliseconds) each antenna spends reading tags. The default value is 20 milliseconds.
Antenna Multiplexing - Enable Antenna Pair	muxEnableAntennaPair	No	Applies only to combined antenna operations. Allowed values are `true` and `false`. `True` means that antennas 1 and 2 are paired, and antennas 3 and 4 are paired. The default value is `false`. See "Antenna Multiplexing Configuration" on page 2-72.
Antenna Multiplexing - Antenna Phase Shift Angle	muxPhaseShiftAngle	No	Applies only to combined antenna operations. Specifies the RF scanning phase shift angle between antennas 1 and 3, and between antennas 2 and 4. The reader firmware supports a RF scanning phase shift angle of 180 degrees; some firmware may support a 90 degree shift angle.

Table 2-21 Tagsys Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
ch 1 Logical Reader Name ch 2 Logical Reader Name ch 3 Logical Reader Name ch 4 Logical Reader Name	ch1LogicalReaderName ch2LogicalReaderName ch3LogicalReaderName ch4LogicalReaderName	No	Specifies the logical reader name for each HF antenna. At least one logical reader name must be specified or the Edge Server will generate an error on startup. Please note that the L100 reader has two antennas and the L200 reader has four antennas.
ch 0 Filter Names ch 1 Filter Names ch 2 Filter Names ch 3 Filter Names	ch1LogicalReaderFilterNames ch2LogicalReaderFilterNames ch3LogicalReaderFilterNames ch4LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.

Antenna Multiplexing Configuration

The Tagsys Medio L100 and L200 readers support combined antenna operations and can manage up to four antennas, organized into two pairs of antennas. Each antenna is able to transmit an RF field, but only one antenna in the pair can receive at a single time. This means that each antenna can transmit an RF field used to read tags, but tag contents can only be read by one antenna of the pair at a given time.

Configuring Tagsys readers to perform read operations on both antennas (or antenna pair) one after the other enhances tag reading performance. The multiplexing mode implements several antenna combinations that are applied one after the other in a loop.

Antenna multiplexing information is organized into patterns. A pattern contains information about which antennas are enabled at the same time as well as the phase shift angle values between the enabled antennas.

If antenna pairs are used, the multiplexing patterns are shown in Table 2-22.

Table 2-22 Multiplexing Patterns With Antenna Pairs

Multiplexing Patterns	PAIR 1 (Antenna 1 + Antenna 2)		PAIR 2 (Antenna 3 + Antenna 4)	
Pattern 1	enabled	disabled	enabled	disabled
Pattern 2	disabled	enabled	disabled	enabled

If no antenna pairs are used, the multiplexing patters are shown in Table 2-23.

Table 2-23 Multiplexing Patterns With No Antenna Pairs

Multiplexing Pattern	Antenna 1	Antenna 2	Antenna 3	Antenna 4
Pattern 1	enabled	disabled	disabled	disabled
Pattern 2	disabled	enabled	disabled	disabled
Pattern 3	disabled	disabled	enabled	disabled
Pattern 4	disabled	disabled	disabled	enabled

For more information on combined antenna operations, see "Combined Antenna Mode" on page 2-64. For detailed multiplexing configuration information, see *Lx00 Platform Firmware Reference Guide* (Tagsys Document Reference 10958D0).

Configuring The MOXA Device

The Tagsys reader will be connected to an IP network with a MOXA serial server. The serial configuration is shown below. If the reader's serial configuration is changed, adjust these settings accordingly.

Baud rate:	38400
Parity:	None
Data bits:	8

Stop bit:	1
Flow control:	None
UART FIFO:	Disable
Interface:	RS-232

A null modem is required if using a straight serial cable (DB9, female-to-male).

ThingMagic

The RFID Edge Server uses two ThingMagic reader drivers:

- ThingMagic Mercury3—Use this driver for the ThingMagic Mercury3 reader.
- ThingMagic Mercury4—Use this driver for the ThingMagic Mercury4 reader.

ThingMagic Mercury3

Table 2-24 lists the ThingMagic Mercury3 reader configuration properties.

Table 2-24 ThingMagic Mercury 3 Reader Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.ThingMagicMercury3PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the ThingMagic reader
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the ThingMagic Reader. The default is 8080.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the default rate times the number of logical readers.

Table 2-24 ThingMagic Mercury 3 Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default is 15000 milliseconds (15 seconds).
Read Timeout	readTimeout	No	Specifies the maximum amount of time (in milliseconds) the reader may spend when servicing a request to read tags in a single antenna's field. The default value is 250 milliseconds.
Write Timeout	writeTimeout	No	Specifies the maximum amount of time (in milliseconds) the reader may spend when servicing a request to program a tag across a single antenna. The default value is 250 milliseconds.
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	An optional Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the “check operation” (verification that there is a single tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation. Note that if the check operation is disabled (this property value is set to `true`) and multiple tags are present in the antenna's field when conducting a tag programming operation, then all tags in the field will be programmed with the same EPC.

Table 2-24 ThingMagic Mercury 3 Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
UHF Antenna 1 Logical Reader Name UHF Antenna 2 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF (915 MHz) antenna. At least one logical reader name must be specified. If one of these properties is left undefined, there will be no logical reader associated with the corresponding antenna, and that antenna will not be accessible via the ALE API. If multiple UHF antennas are assigned the *same* logical reader name, then the driver will treat them as a single combined antenna. The lowest-numbered antenna in the combined antenna grouping is checked to set antenna-specific properties and used for tag programming.
UHF Antenna 1 Filter Names UHF Antenna 2 Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
HF Antenna 1 Logical Reader Name HF Antenna 2 Logical Reader Name	hf1LogicalReaderName hf2LogicalReaderName	Must configure one	Specifies the logical reader name for each HF (13.56 MHz) antenna. At least one logical reader name must be specified. If one of these properties is left undefined, there will be no logical reader associated with the corresponding antenna, and that antenna will not be accessible via the ALE API.
HF Antenna 1 Filter Names HF Antenna 2 Filter Names	hf1LogicalReaderFilterNames hf2LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.

ThingMagic Mercury4

Table 2-25 lists the ThingMagic Mercury4 reader configuration properties.

Table 2-25 ThingMagic Mercury 4 Reader Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	Must be set to: `com.connecterra.ale.readertypes.ThingMagicMercury4PhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the ThingMagic reader
Reader Port	port	No	The TCP port the Edge Server will use when establishing connections to the ThingMagic Reader. The default is 8080.
Default Rate	defaultRate	Yes	The period (in milliseconds) between the start of one read cycle and the start of the next. Note that if multiple logical readers are simultaneously active, then each logical reader will be read at an interval equal to the default rate times the number of logical readers.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default is 15000 milliseconds (15 seconds).
Read Timeout	readTimeout	No	Specifies the maximum amount of time (in milliseconds) the reader may spend when servicing a request to read tags in a single antenna's field. The default value is 250 milliseconds.
Write Timeout	writeTimeout	No	Specifies the maximum amount of time (in milliseconds) the reader may spend when servicing a request to program a tag across a single antenna. The default value is 250 milliseconds.

Table 2-25 ThingMagic Mercury 4 Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Disable Programming Cycle Check	disableProgrammingCycleCheck	No	An optional Boolean property (allowed values are `true` and `false`) specifying whether the driver disables the "check operation" (verification that there is a single tag in an antenna's field prior to conducting a tag programming operation). The default value is `false`, meaning the driver conducts the check operation. Note that if the check operation is disabled (this property value is set to `true`) and multiple tags are present in the antenna's field when conducting a tag programming operation, then all tags in the field will be programmed with the same EPC.
UHF Antenna 1 Logical Reader Name UHF Antenna 2 Logical Reader Name UHF Antenna 3 Logical Reader Name ... UHF Antenna 8 Logical Reader Name	uhf1LogicalReaderName uhf2LogicalReaderName uhf3LogicalReaderName ... uhf8LogicalReaderName	Must configure one	Specifies the logical reader name for each UHF (915 MHz) antenna. At least one logical reader name must be specified. If one of these properties is left undefined, there will be no logical reader associated with the corresponding antenna, and that antenna will not be accessible via the ALE API. If multiple UHF antennas are assigned the *same* logical reader name, then the driver will treat them as a single combined antenna. The lowest-numbered antenna in the combined antenna grouping is checked to set antenna-specific properties and used for tag programming.

Table 2-25 ThingMagic Mercury 4 Reader Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
UHF Antenna 1 Filter Names UHF Antenna 2 Filter Names UHF Antenna 3 Filter Names ... UHF Antenna 8Filter Names	uhf1LogicalReaderFilterNames uhf2LogicalReaderFilterNames uhf3LogicalReaderFilterNames ... uhf8LogicalReaderFilterNames	No	A blank-separated list of filter names for each logical reader. This value is defined in the Administration Console or in the `edge.props` file.
N/A	rfProtocols	No	See rfProtocols. Used only for backward compatibility with releases prior to RFTagAware 1.3.

List of Protocols	rfProtocols	No	A blank-separated list of RF protocols. Mercury4 is a multi-protocol reader and can operate in read-only or read-write mode. At least one protocol must be specified. At present, the valid values are:
			`epcClass0`
			`epcClass1`
			`epcClass1Gen2`
			`ALL`
			The default value is `ALL`.

The ThingMagic reader obtains its IP network configuration dynamically via DHCP, or statically through one of the reader configuration interfaces. Refer to the *ThingMagic User Guide* for further details.

Zebra

This section describes driver configuration information for the Zebra R110XiIIIPlus, R110Xi and R4MPlus label printers.

- The R110XiIIIPlus printer supports the writing (programming) of Class 0+ and Class 1 tags embedded within label stock (“smart labels”) and the printing of those labels.

- The R110Xi printer supports the writing and printing of Class 0+ and Class 1 tags embedded within label stock (“smart labels”).
- The R4MPlus printer does the same for Class 1 tags.

The Zebra R110XiIIIPlus, R110Xi and R4MPlus printer drivers share all the same properties except `class`, as shown in the following table

Table 2-26 Zebra Configuration Properties

Field Label	Property Name	Required?	Property Value and Description
Device Type	class	Yes	R110XiIIIPlus: `com.connecterra.ale.readertypes.ZebraR110XiIIIPlusPhysicalReader` R110Xi: `com.connecterra.ale.readertypes.ZebraR110XiIIIPlusPhysicalReader` R4MPlus: `com.connecterra.ale.readertypes.ZebraR4MPlusPhysicalReader`
Reader Hostname	hostname	Yes	The DNS name or IP address of the printer’s LAN adapter.
Reader Port	port	No	The TCP port the Edge Server will target when establishing connections to the printer’s LAN adapter. The default value is 9100, the Zebra R110XiIIIPlus device’s factory default.
Socket Timeout	socketTimeout	No	The TCP socket timeout interval (milliseconds). The default value is 15000 milliseconds (15 seconds).
Logical Reader Name	uhf1LogicalReaderName	Yes	The logical reader name assigned to the Zebra device’s single integrated UHF antenna.
Enable RFID Encoding	programRFID	No	A Boolean (allowed values are `true` and `false`) specifying whether the reader driver instructs the printer to program EPC data into an embedded RFID tag. The default value is `true`.

Table 2-26 Zebra Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
ZPL Form	zplFilename	No	Specifies the pathname of a file containing Zebra ZPL commands defining a smart label form. Typically, the PCSpec `readerParameters` attribute specifies ZPL-based label design; the reader parameter name is `zplScript`. This property provides a mechanism for specifying a default label design in the event the `PCSpec` does not contain ZPL reader parameters. When present, the ZPL Form refers to a text file (`label.zpl`) containing the set of ZPL commands specifying the smart label layout and dynamic (variable) content. See "Using the Zebra Printing Language (ZPL II)" on page 2-82. **Note:** If specified, this property must point to a valid ZPL file.
Retries	retries	No	Defines the number of labels to retry in case of a failed RFID operation. The valid range is 1 through 10, the default value is 3.

Error Handling	errorHandling	No	Specifies an error handling operation, if an RFID operation fails after the number of retries (specified in the `retries` property). The valid values are:
			
			`NO_ACTION`
			`PAUSE_MODE`
			`ERROR_MODE`
			
			The default value is `NO_ACTION`.

Table 2-26 Zebra Configuration Properties (Continued)

Field Label	Property Name	Required?	Property Value and Description
Transponder Position	transponderDisplacement	No	Applies only to R4MPlus printer. Specifies the value of parameter in the `^RS` command issued to the printer. Use this property only if necessary. Refer to *Zebra R4MPlus User Guide* for explanation and usage guidelines
Verify Pre-encoded A5A5	verifyValidData	No	Applies only to R4MPlus printer. A Boolean (allowed values are `true` and `false`). The default value is `false`. When `true`, the printer fails the tag programming operation if the first two bytes of the tag are not `0xA5A5`.

Using the Zebra Printing Language (ZPL II)

ZPL is a scripting language for specifying the layout and contents of printed labels. ZPL is described in detail in the Zebra document, *ZPLII Programming Guide (Zebra Document 45541LB-R3)*.

The Zebra printer driver sends its printer a separate collection of ZPL commands with each "smart label" (a printed label with an embedded RFID tag) to be printed and programmed. Users specify smart label ZPL commands when defining a PCSpec (see *Programming with the ALE and ALEPC APIs*). A PCSpec `readerParameters` field carries the ZPL script as a String object; the reader parameter name is `zplScript`. As an alternative to specifying ZPL within a PCSpec, the Edge Server administrator may specify a default ZPL script using the Zebra reader driver *ZPL Form* property. This property provides a mechanism for specifying ZPL commands that serve as a default label design in the event the PCSpec `readerParameters` field does not carry a key/value pair containing a ZPL script.

Below is a sample ZPL script:

```
^XA
^FX *** NOTE: Lines, beginning with ^FX are comments ***^FS
^FX Set home position ^FS
^LH30,35
```

```
^FX Write EPC HEX ^FS
^WT,,,,1^FD[EPC_HEX]^FS


^FX "Ship From" text ^FS
^FO20,15^A0N,18,,^FDSHIP FROM:^FS
^FO20,40^A0N,26,,^FDAcme^FS
^FO20,65^A0N,26,,^FDCorporation^FS
^FO20,90^A0N,26,,^FDP.O. Box 61900^FS
^FO20,115^A0N,26,,^FDDallas, TX 75261^FS


^FX Draw vertical line ^FS
^FO300,0^GB0,180,2^FS


^FX "Ship to" text ^FS
^FO320,15^A0N,18,,^FDSHIP TO:^FS
^FO320,40^A0N,26,,^FDRetailer Distribution Center^FS
^FO320,65^A0N,26,,^FD200 Main Street^FS
^FO320,90^A0N,26,,^FDSpringfield, MA01103^FS


^FX Draw horizontal line ^FS
^FO0,180^GB720,0,4^FS


^FX Postal code text ^FS
^FO20,195^A0N,18,,^FDSHIP TO POSTAL CODE:^FS
^FO30,275^A0N,32,,^BC,90,,Y,^FD(420) 01103^FS


^FX Draw vertical line ^FS
^FO340,180^GB0,270,2^FS


^FX Carrier text ^FS
^FO360,195^A0N,18,,^FDCARRIER:^FS
^FO360,245^A0N,36,,^FDAcme Freightways^FS
```

```
^FO360,295^A0N,30,,^FDPRO: 1234^FS
^FO360,345^A0N,30,,^FDB/L: 5678^FS


^FX Draw horizontal line ^FS
^FO0,450^GB720,0,4^FS


^FX EPC text ^FS
^FO0,465^A0N,32,,^FDEPC:^FS
^FO20,575^A0N,32,,^FD[EPC_TAG_URI]^FS


^FX Draw horizontal line ^FS
^FO0,670^GB720,0,4^FS


^FX SKU text ^FS
^FO0,685^A0N,32,,^FDSKU:^FS
^FO15,724^A0N,36,,^FDABC21270^FS


^FX Draw hoarizontal line ^FS
^FO0,785^GB720,0,4^FS


^FX GTIN text ^FS
^FO0,800^A0N,32,,^FDGTIN^FS
^FO100,890^A0N,32,,^BC,90,,Y,^FD>; (01) 10036000212706 ^FS
^XZ
```

If you want to print a specified EPC value on the smart label, the user-supplied ZPL must include special substitution strings, indicating in what representation the tag values are to be printed. The RFID Edge Server recognizes the following substitution string values:

[EPC_HEX]	Will be replaced by the hexadecimal representation of the tag value.
[EPC_ID_URI]	Will be replaced by the pure identity URN representation of the tag value.
[EPC_TAG_URI]	Will be replaced by the tag URN representation of the tag value.

Example:

```
^XA
^FO20,575^A0N,32,,^FD[EPC_TAG_URI]^FS
^XZ
```

General Troubleshooting

The section applies to all readers.

Problem: While an active subscription is running, the reader frequently displays a failed connection message.

Possible reason: Verify that the *socketTimeout* field is blank (the default setting is 15000 ms). You can change this value, but it is impractical to set it lower than 5000.

Problem: You see unusually high CPU utilization when running an active subscription.

Possible reason: Verify that the *defaultRate* value is a **non-zero** positive number.

CHAPTER 3

Configuring and Controlling Stack Lights

A stack light is a device containing up to five signaling lights that can be attached to an RFID device. WebLogic RFID Edge Server supports multi-color stack light configuration for these devices: Symbol AR400, Symbol XR400, and Alien ALR-9780.

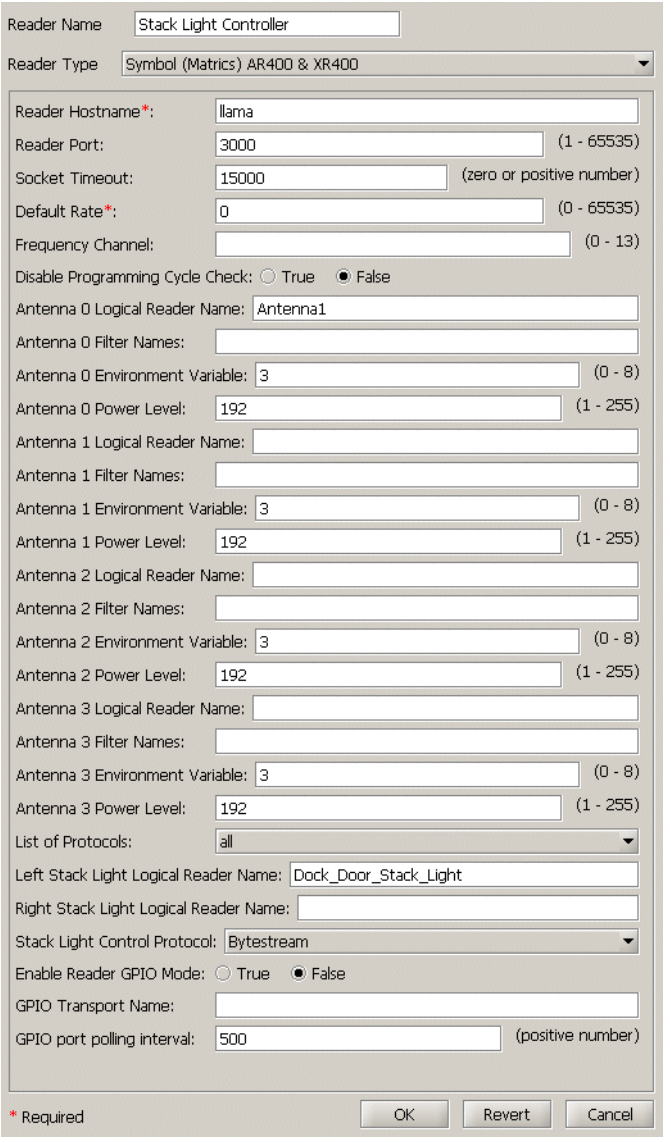
The following sections describes configuring and controlling stack lights.

Stack Light Configuration

For each stack light to be used, you assign a Logical Reader Name during reader configuration (much like an antenna). Other properties may be configured as needed.

Figure 3-1 illustrates a Symbol AR400 reader stack light configuration, showing example host names and logical reader names, and default values for other properties.

Figure 3-1 Symbol AR400 Stack Light Configuration



Stack Light Control

Stack light control can be accomplished by writing a special EPC value to the logical reader representing a stack light. The EPC is represented by the URI:
`urn:connecterra:stacklight:update=<update value>`

The *update value* consists of five hexadecimal digits. Each digit sets the display on one of the five supported colors (from most significant to least significant): White, Blue, Green, Amber, and Red. Each digit can be one of the following values:

- 0 = turns the color off
- 1-8 = turns the color on for 5 seconds times the value shown
- 9 = turns the color on indefinitely, until a different value is written
- F = does not change the current state

Values of A-E are reserved for future use.

If the update value written to the logical reader has more than five digits, the rightmost five digits are read to set the stack light colors (e.g., 88880000 is read as 80000).

Note: For specific devices, an exception is thrown when the update value has more than five digits; use an update value containing only five digits in those cases.

If the update value has less than five digits, the number is padded with zeroes on the left before the value is sent to the logical reader (e.g., FF0 is read as 00FF0).

Example Value: A URI of `urn:connecterra:stacklight:update=092F0` written to a stack light configured as described above will perform the following actions on the stack light:

- 0 – turns off the White light
- 9 – turns on the Blue light indefinitely, until next update
- 2 – turns on the Green light for 10 seconds
- F – leaves the Amber light unchanged
- 0 – turns off the Red light

Listing 3-1 Example API call

```
// This is a code fragment
ALEPC alePCClient = new AxisALEPCClient(<alepcServiceURL>);
ALEPCFactory alePCFactory = AxisALEPCFactory.getInstance();
PCSpec pcSpec = alePCFactory.createPCSpec();
pcSpec.addLogicalReaderName("Dock_Door_Stack_Light");
alePCClient.immediate(
              pcSpec, new URI("urn:connecterra:stacklight:update=092F0"));
```

Alternatively, stack light control can be accomplished by writing a value to a tag memory URI:

`urn:connecterra:tagmem:@stacklight.X.Y` and the update value is `urn:epc:raw:<update value>`, where:

- `X`—Number of bits (can be 4, 8, 12, 16, or 20); the memory bank is 20 bits wide, each 4 bits corresponds to one light.
- `Y`—Offset (can be 0, 4, 8, 12, or 16); an offset of 12 means that the first twelve bits are skipped.
- `update value`—Up to five hexadecimal digits (as described in "Stack Light Control" on page 3-3), depending on the number of bits and the offset value specified by `X.Y`.

Examples:

- `urn:connecterra:tagmem:@stacklight.4.12` and update value `urn:epc:raw:64.x9`; turns the amber light on indefinitely (starting from bit 12, 4 bits).

  In this example, `@stacklight.4.12` means write 4 bits starting at offset 12. Offsets start at 0 and 12 bits are skipped. The 4 bits in the update value (for example, x9) are written in bits 12, 13, 14, 15.

- `urn:connecterra:tagmem:@stacklight.20.0` and update value `urn:epc:raw:64.x99999`; turns on all the lights indefinitely (starting from bit 0, 20 bits).

Listing 3-2 Example PCSpec

```
<?xml version="1.0" encoding="UTF-8"?>
<PCSpec xmlns="http://schemas.connecterra.com/alepc">
```

```
  <applicationData>Write Tag Memory with stacklight</applicationData>
  <applicationData>pass EPC value as an argument</applicationData>
  <logicalReaders>
    <logicalReader>StackLight</logicalReader>
  </logicalReaders>
  <boundarySpec>
    <trials>1</trials>
    <duration>4000</duration>
  </boundarySpec>
  <accessSpec>
    <operations>
      <operation>
      <!-- Write Tag Memory with stacklight;pass EPC value as argument -->
            <write field="urn:connecterra:tagmem:@stacklight.20.0">
                <param>epc</param>
        </write>
      </operation>
    </operations>
  </accessSpec>
</PCSpec>
```

For more information, see Writing Tags Using the ALEPC API in *Programming with the ALE and ALEPC APIs*.

CHAPTER 4

Controlling RFID Devices via PLC

The following sections describe how the RFID Edge Server communicates with and controls RFID devices by interacting with a programmable logic controller (PLC).

Overview of PLC Communications

The WebLogic RFID Edge Server allows you to receive information about other devices and control them, by communicating with a *programmable logic controller* (PLC), which is a specialized industrial computer used for automation of real-world processes. A PLC has a number of registers, which are used to hold information. Registers are sometimes referred to as items or tags (not to be confused with EPC tags). Applications interacting with the PLC can either:

- read the contents of one or more registers (referred to as *inbound messages*) and take actions based on the values read, or
- write information to one or more registers to make it available to other applications (referred to as *outbound messages*).

Configuring the Edge Server for PLC Communications

Configuring the RFID Edge Server to enable PLC communications requires adding the following items to the `edge.props` file:

- One or more *transports*, which specify the underlying mechanism used to send messages and the location to which messages should be sent. There may be other configuration options specific to the transport chosen. The available implementations are `OPC_XML`, `StarThis`, `ModBus` and `Reader` (replaces `GPIO`, deprecated in this release).
- One or more *message conventions*, which must correspond to the conventions expected by the PLC with which the Edge Server communicates. The conventions supported are `Simple`, `CounterPair`, and `Multiple`. More than one message can use the same message convention.
- One or more *messages*, which specify the registers on the PLC to read from or write to, and the values to be written. The same message can be sent for multiple events. The messages supported are `inboundMessage` and `outboundMessage`.

In general, users who want to enable their installation of the RFID Edge Server to control RFID devices using programmable logic controllers will need some custom development effort. The examples provided below illustrate possible configurations after custom development work has been completed.

PLC-Connected Reader and Printer Example

Note: All property names shown in this section should be prefixed with `com.connecterra.ale.plc`, which is abbreviated `[CCAP]` in the properties below.

1. Create a backup copy of `edge.props` and open the original for editing.
2. Define a transport to be used for PLC communications. The supported implementations are `OPC_XML`, `StarThis`, and `ModBus`.

```
# ModBus configuration
[CCAP].plcTransport.modbus1.metaName = ModBus
[CCAP].plcTransport.modbus1.hostname = ip_address
[CCAP].plcTransport.modbus1.socketTimeout = 10000
[CCAP].plcTransport.modbus1.pollInterval = 500

# OPC_XML configuration
[CCAP].plcTransport.opcABC.metaName = OPC_XML
[CCAP].plcTransport.opcABC.hostname = http://plchost/path

# StarThis configuration
[CCAP].plcTransport.ab1756.metaName = StarThis
[CCAP].plcTransport.ab1756.licenseDir = ..\licenses
[CCAP].plcTransport.ab1756.storageDir = ..\var\plc-storage
```

```
[CCAP].plcTransport.ab1756.hostname = ab1756
[CCAP].plcTransport.ab1756.backplanePort = 0
[CCAP].plcTransport.ab1756.pollInterval = 50
```

3. Choose a message convention. The types supported are `Simple`, `CounterPair`, and `Multiple`. The notations `<var1>` and `<var2>` should be replaced by property names that you choose.

```
# SimpleTransaction configuration
[CCAP].plcMessageConvention.<var1>.metaName = Simple
# The plcTransport name shown below should be a hostname from step 2.
[CCAP].plcMessageConvention.<var1>.plcTransport = ab1756

# CounterPairTransaction configuration
[CCAP].plcMessageConvention.<var2>.metaName = CounterPair

# Define ModBus PLC message convention for writing
[CCAP].plcMessageConvention.multiple.metaName = Multiple
[CCAP].plcMessageConvention.multiple.plcTransport = modbus1

# The plcTransport name shown below should be a hostname from step 2.
[CCAP].plcMessageConvention.<var2>.plcTransport = ab1756
[CCAP].plcMessageConvention.<var2>.restart.receiveCounterItem =
PLCRestart[2]
[CCAP].plcMessageConvention.<var2>.restart.ackCounterItem =
PLCRestartAck[2]
[CCAP].plcMessageConvention.<var2>.restart.dataItems = PLCRestart[0]
[CCAP].plcMessageConvention.<var2>.restart.prefetch = true

# Configure the next two properties on ONE client only
[CCAP].plcMessageConvention.<var2>.restartAckStatusItem = PLCRestartAck[0]
[CCAP].plcMessageConvention.<var2>.restartDelay = 100
```

Note: The array elements shown above and in step 4 refer to PLC registers on the PLCs in use at your organization. Both the register names and element numbers may differ from the sample information shown.

4. Define the inbound and outbound messages for the message convention chosen in the prior step. The notations `<var1>` and `<var2>` should be replaced by the property names you chose in step 3.

```
# Simple inboundMessage configuration
[CCAP].inboundMessage.doorOpen.plcMessageConvention = <var1>
```

```
[CCAP].inboundMessage.doorOpen.receiveItem = D2052
[CCAP].inboundMessage.doorOpen.matchValues = true

# CounterPair inboundMessage configuration
[CCAP].inboundMessage.BCRArrival.plcMessageConvention = <var2>
[CCAP].inboundMessage.BCRArrival.receiveCounterItem = BCRArrival[19]
[CCAP].inboundMessage.BCRArrival.ackCounterItem = BCRArrivalAck[0]
[CCAP].inboundMessage.BCRArrival.dataItems = BCRArrival

# CounterPair outboundMessage configuration
[CCAP].outboundMessage.SgtinAssign.plcMessageConvention = <var2>
[CCAP].outboundMessage.SgtinAssign.sendCounterItem = SgtinAssign[59]
[CCAP].outboundMessage.SgtinAssign.ackCounterItem = SGTINAssignAck[0]
[CCAP].outboundMessage.SgtinAssign.dataItems = SgtinAssign
```

5. Save the changes to `edge.props` and restart the Edge Server.

6. Configure your RFID devices to communicate with the PLC by defining devices as shown below using the **RFID Devices** node in the Administration Console:

- **Device Type**: PLC-Connected Barcode Reader
  Logical Reader Name: <a reader name you choose>
  Inbound PLC Message Name: <must match the name of the inboundMessage carrying the bar code; for example, `BCRArrival` in step 4>

- **Device Type**: PLC-Connected Label Print & Apply
  Logical Reader Name: <a reader name you choose>
  Outbound PLC Message Name: <must match the name of the outboundMessage; for example, `SgtinAssign` in step 4>

PLC Stack Light Example

For a PLC stack light, there are two components that you must configure:

- PLC stack light device (a PLC stack light logical reader)
- PLC outbound message definition

You can configure the PLC stack light device in the Administration Console or `edge.props` file. The PLC outbound message can only be defined in `edge.props`.

In order to map which PLC outbound message the PLC stack light device uses, you specify the PLC outbound message name, defined in `edge.props` (in step 3 below, `[CCAP].outboundMessage.`**`stacklight`**`.plcMessageConvention = multiple`) when you

configure the PLC stack light device in the Administration Console (in step 5 below, PLC set stack light message name = **stacklight**). The PLC outbound message name (for example, stacklight) can be any string you choose, but it must be specified in both the PLC stack light device configuration and the PLC outbound message definition.

1. Create a backup copy of `edge.props` and open the original for editing.
2. Define a PLC stack light device.

```
#Define ModBus PLC stack light reader. Note that the stack light reader is
#defined either using the Administration Console or by editing the
#edge.props file. The default behavior
#com.connecterra.ale.dynamicConfig.enabled = true, means that you configure
#the reader using the Administration Console. If you want to define and
#configure the reader in the edge.props file, set this property to false.
com.connecterra.ale.reader.stacklight.class =
com.connecterra.ale.readertypes.PLCStackLightPhysicalReader
com.connecterra.ale.reader.stacklight.plcMessage = stacklight
com.connecterra.ale.reader.stacklight.defaultRate = 0
com.connecterra.ale.reader.stacklight.stackLightLogicalReaderName =
StackLight
```

3. Choose a message convention and define the outbound messages for the message convention.

```
# Multiple outboundMesssage configuration for 'stacklight'
com.connecterra.ale.plc.outboundMessage.stacklight.plcMessageConvention =
multiple
com.connecterra.ale.plc.outboundMessage.stacklight.items = c1 c2 c3 c4 c5
```

4. Save the changes to `edge.props` and restart the Edge Server.
5. Configure your RFID devices to communicate with the PLC by defining devices as shown below using the **RFID Devices** node in the Administration Console:

- **Note:** If you configured the stack light reader in `edge.props`, do not perform this configuration step in the Administration Console.
  Device Type: PLC Stack Light
  PLC Stack Light Logical Reader Name: <a reader name you choose>
  PLC set stack light message name: <must match the name of the outboundMessage; for example, `stacklight` in step 3>
  Enable PLC command optimization: `false`

CHAPTER 5

Triggers

This chapter provides the following sections:

Introduction to Event Triggers

Applications can define event cycle specifications (`ECSpec`) and programming cycle specifications (`PCSpec`) where the beginning or end of each cycle is triggered by external events. BEA provides an extensible mechanism for connecting sources of external events to the ALE engine.

OLE for Process Control (OPC) Trigger Driver

OPC is a series of standards specifications that define a standard set of objects, interfaces, and methods for use in process control and manufacturing automation applications to facilitate interoperability. (For more information on OPC, see `http://www.opcfoundation.org`.)

WebLogic RFID Edge Server includes a driver for OPC triggers. An event or programming cycle in the Edge Server can be triggered by polling for a change in an OPC item. The Edge Server

communicates with the OPC service using the OPC XML-DA protocol, which is a SOAP interface to an OPC Data Access provider. The OPC XML-DA implementation is provided by a third party.

The general form of the trigger URI is:

```
opcpoll:itemName=item;http://hostname/location
```

where:

- `item` is the name of the OPC item that the driver polls for changes.
- `http://hostname/location` is the URL that is used to create the connection to the OPC XML-DA server.

When an event cycle or programming cycle that uses an OPC trigger is first requested (that is, when the event cycle or programming cycle is invoked using the `poll` or `immediate` method or subscribed using `subscribe`), the Edge Server polls the OPC XML-DA server for the current value of the specified OPC item. Subsequently, each time the value of the OPC item changes, a trigger will be delivered to the event cycle or programming cycle.

TCP Trigger Driver

WebLogic RFID Edge Server provides a TCP trigger driver that uses simple Telnet input to start or stop ALE event cycles in the Edge Server.

The following example shows how to invoke TCP triggers manually:

1. Open two Telnet sessions, one for the start trigger and one for the stop trigger.

   For example,

   ```
   :prompt> telnet localhost 7070
   :prompt> telnet localhost 7071
   ```

2. In the Administration Console, edit an ECSpec to set the start and stop trigger URIs, then click **Deploy**.

   a. **Start Trigger URI**: `tcplisten:`*`ip_address`*`:7070`

   b. **Stop Trigger URI**: `tcplisten:`*`ip_address`*`:7071`

   Note: If you specify two colons, for example, `tcplisten::7070`, it uses all IP addresses.

3. Activate the ECSpec by clicking **Test** or **Activate Once**.

4. Invoke the start trigger by pressing a carriage return in the Telnet session `localhost:7070`.

When the start trigger is fired, the event cycle starts receiving tag data. The event cycle stops when either the stop trigger fires or other stop conditions are met.

5. Invoke the stop trigger by pressing a carriage return in the Telnet session `localhost:7071`.

   The event cycle terminates and an ECSpec report is generated.

In production, an application invokes a TCP trigger by:

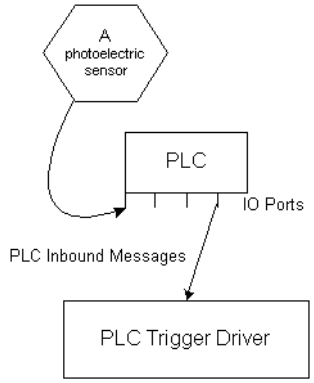
1. Opening a socket connection to the TCP port (in the previous example, `localhost = 7070`)
2. Writing a line feed (a character code that advances the screen cursor or printer to the next line, `<LF>` or `0x0A` hex)

   The TCP trigger driver listens for the line feed. When supplied, it provides the trigger URI to start or stop the event cycle.

PLC Trigger Driver

The RFID Edge Server PLC trigger driver converts PLC messages into trigger URIs for starting or stopping ALE event cycles. Figure 5-1 illustrates a typical PLC trigger setup.

Figure 5-1 PLC Trigger Input



For more information on how the RFID Edge Server communicates with and controls RFID devices by interacting with PLCs, see "Controlling RFID Devices via PLC" on page 4-1.

Configuring PLC Triggers

You configure PLC triggers in the `edge.props` file using the following steps:

Note: All property names shown in this section should be prefixed with `com.connecterra.ale.plc`, which is abbreviated `[CCAP]` in the properties below.

1. Create a backup copy of `edge.props` and open the original for editing.
2. Define a transport to be used for PLC communications; for example, a `Modbus` transport named *modbus0*.

```
# ModBus transport configuration
[CCAP].plcTransport.modbus0.metaName = ModBus
[CCAP].plcTransport.modbus0.hostname = plcHostName
[CCAP].plcTransport.modbus0.socketTimeout = 10000
[CCAP].plcTransport.modbus0.pollInterval = 10000
```

3. Define a message convention; for example, the `Simple` message convention named *simplemodbus0*.

```
[CCAP].plcMessageConvention.simplemodbus0.metaName = Simple
[CCAP].plcMessageConvention.simplemodbus0.plcTransport = modbus0
```

4. Define the inbound message for the message convention chosen in step 3.

```
#Define inboundMessage 'photoeye1' which is connected to a PLC pin with
#address 2052. ('d' is a convention referring to discrete input in the
#ModBus protocol.)
[CCAP].inboundMessage.photoeye1.plcMessageConvention = simplemodbus0
[CCAP].inboundMessage.photoeye1.receiveItem = d2052
[CCAP].inboundMessage.photoeye1.matchValues = true
```

5. Define the PLC trigger.

```
#Define a trigger named 'plcmessage' using the PLCTriggerDriver
com.connecterra.ale.triggerDriver.plcmessage.class=com.connecterra.ale.tri
ggertypes.PLCTriggerDriver
```

6. Save the changes to `edge.props` and restart the Edge Server.
7. In the Administration Console, edit an ECSpec to set a start or stop condition trigger URI; for example, **Start Trigger URI**: *plcmessage:photoeye1*

Directional Trigger Driver

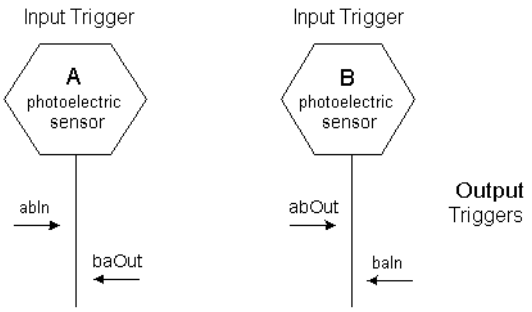
WebLogic RFID Edge Server includes a directional trigger driver that gets input from two source points, A and B, and determines if an object is moving from A to B or from B to A.

How It Works

In the following example, two photoelectric sensors are positioned at a dock door, one at reading point A and one at reading point B. Tagged items move between points A and B *in one direction at a time*.

Figure 5-2 shows the input triggers, A and B, and the directional output triggers.

Figure 5-2 Input and Output Triggers

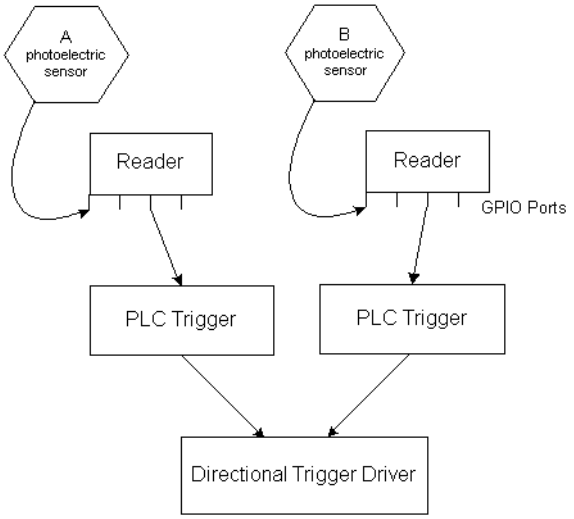


Moving items which first block photoelectric sensor A (generating output trigger abIn), then block sensor B (generating output trigger abOut), signal that AB traversal is finished. Moving items which block photoelectric sensor B first, then sensor A, (generating output triggers baIn then baOut), signal that BA traversal is finished.

Note: There is a maximum allowed travel time (T) between point A and point B. The trigger driver resets to the idle state after travel time T elapses, even if an object moves past photoelectric sensor A (moves in) but fails to move past photoelectric sensor B at the exit point (does not move out).

The directional trigger driver listens for input triggers. Input triggers might be PLC triggers, whose sources are PLC inbound messages which receive inputs from photoelectric sensors or other IO inputs, such as GPIO. Other types of triggers can also provide inputs, such as TCP triggers.

Figure 5-3 Input Sources



The directional trigger driver monitors the status (state change) of the input triggers, for example, when the photoelectric sensors are blocked. When the trigger driver receives the input trigger status, it generates trigger messages for trigger listeners in the event cycle, one listener for event start conditions and one for event stop conditions. When the start trigger is fired, the event cycle starts getting tag data; the event cycle stops when either the stop trigger fires or the cycle duration time expires.

Configuring the Directional Trigger Driver

You configure the directional trigger driver in the `edge.props` file by specifying the trigger driver class name, the timeout (in milliseconds), and two input triggers.

In the `edge.props` file example in Listing 5-1, the directional trigger name is *door1*; you can substitute the trigger name with any string.

Listing 5-1 Directional Trigger Driver Configuration

```
#Define the directional trigger
#Trigger driver class name
com.connecterra.ale.triggerDriver.door1.class=com.connecterra.ale.triggertypes
.DirectionTriggerDriver
```

```
#Directional trigger state timeout (if start trigger is fired, but stop
#trigger fails to fire, the directional trigger will reset to idle state
#after timeout period in ms)
com.connecterra.ale.triggerDriver.door1.timeout=10000

#input trigger URI at point 'a'
com.connecterra.ale.triggerDriver.door1.aTrigger=plcmessage:photoeye1

#input trigger URI at point 'b'
com.connecterra.ale.triggerDriver.door1.bTrigger=plcmessage:photoeye2
```

The directional trigger driver generates four output trigger URIs; for example, *door1*:abIn, *door1*:abOut, *door1*:baIn, *door1*:baOut. You use the trigger URI to specify the starting condition (for example, *door1*:abIn) or stopping condition (for example, *door1*:abOut) in the event cycle definition.

Note: abIn, abOut, baIn, and baOut are defined names and cannot be changed.

Examples

The following examples illustrate two directional trigger configurations.

Using Reader GPIO Ports

This example describes using the directional trigger driver with a Symbol XR400 reader (it does not work with the AR400 reader) and Symbol XR400 Reader Light Indication Box. Refer to the reader manufacturer's manual for the wiring details (Symbol Reader Manual, *XR Series RFID Reader Integrator Guide*, Part Number 72E-71773-03 Rev. A, February 2006).

Photoelectric sensors are connected to the Light Indication Box, then to the reader GPIO input ports (at pins #4 and #5). The GPIO port status is forwarded to the directional trigger driver using the Reader type PLC transport and the simple message convention.

Directional Trigger and Reader Configuration: Main Steps

To configure the directional trigger and the reader, perform the following steps:

1. In the edge.props file, define the following properties (see Listing 5-2):
   - GPIO transport name and type
   - Message convention
   - Input triggers for points a and b

  - Directional trigger driver

2. In the ECSpec (for the example directional trigger named *door1*), define the following start and stop conditions:
   - Start Trigger URI: `door1:abIn` for ‘a’ to ‘b’ movement (or `door1.baIn` for ‘b’ to ‘a’ movement)
   - Stop Trigger URI: `door1:abOut` for ‘a’ to ‘b’ movement (or `door1.baOut` for ‘b’ to ‘a’ movement)
   - Stop duration (for example, 10000 ms)
3. Configure the Symbol XR400 reader for GPIO mode, as described in “Enabling GPIO Mode (Symbol XR400 Reader Only)” on page 2-62.
4. In the Administration Console, configure these Symbol XR400 reader parameters:
   - **Enable Reader GPIO Mode** (`enableGPIO`): `true`.
   - **GPIO Transport Name**: `mygpio` (as defined in `edge.props`, bolded in Listing 5-2)

Listing 5-2 Example GPIO Directional Trigger Configuration

```
#In edge.props file
#Define the PLC Transport type ‘Reader’ named ‘mygpio’
com.connecterra.ale.plc.plcTransport.mygpio.metaName = Reader

#Define the message convention ‘Simple’ named ‘simplegpio’
com.connecterra.ale.plc.plcMessageConvention.simplegpio.metaName = Simple
com.connecterra.ale.plc.plcMessageConvention.simplegpio.plcTransport = mygpio

#Define inboundMessage named ‘photoeye1’
com.connecterra.ale.plc.inboundMessage.photoeye1.plcMessageConvention =
simplegpio

#Photoeye1 connected to the reader’s GPIO input port pin #5
com.connecterra.ale.plc.inboundMessage.photoeye1.receiveItem = input5
com.connecterra.ale.plc.inboundMessage.photoeye1.matchValues = true

#Define inboundMessage named ‘photoeye2’
com.connecterra.ale.plc.inboundMessage.photoeye2.plcMessageConvention =
simplegpio

#Photoeye2 connected to reader’s GPIO input port pin #4
com.connecterra.ale.plc.inboundMessage.photoeye2.receiveItem = input4
com.connecterra.ale.plc.inboundMessage.photoeye2.matchValues = true
```

```
#Define the input trigger named 'plcmessage' using the PLCTriggerDriver
com.connecterra.ale.triggerDriver.plcmessage.class=com.connecterra.ale.trigger
types.PLCTriggerDriver

#Define the directional trigger
#Driver class name
com.connecterra.ale.triggerDriver.door1.class=com.connecterra.ale.triggertypes
.DirectionTriggerDriver

#Directional trigger state timeout (if start trigger is fired, but stop
#trigger fails to fire, the trigger will reset to idle state after timeout
#period in ms)
com.connecterra.ale.triggerDriver.door1.timeout=10000

#input trigger URI at point 'a'
com.connecterra.ale.triggerDriver.door1.aTrigger=plcmessage:photoeye1

#input trigger URI at point 'b'
com.connecterra.ale.triggerDriver.door1.bTrigger=plcmessage:photoeye2
```

Using a ModBus PLC

The following example defines a directional trigger named *door2* and uses two PLC IO inputs as input triggers. The PLC inputs are forwarded to the directional trigger driver using the `ModBus` type PLC transport and the `simple` message convention. The RFID Edge Server communicates with the PLC using the Modbus protocol. See "Controlling RFID Devices via PLC" on page 4-1.

The four output trigger URIs are *door2*`:abIn`, *door2*`:abOut`, *door2*`:baIn`, *door2*`:baOut`.

Listing 5-3 Example Directional Trigger Configuration Using PLC Modbus Protocol

```
#In edge.props file
#Define the ModBus transport named 'modbus0'
com.connecterra.ale.plc.plcTransport.modbus0.metaName = ModBus
com.connecterra.ale.plc.plcTransport.modbus0.hostname = localhost
com.connecterra.ale.plc.plcTransport.modbus0.socketTimeout = 10000
com.connecterra.ale.plc.plcTransport.modbus0.pollInterval= 1000

#Define the message convention 'Simple'
com.connecterra.ale.plc.plcMessageConvention.simplemodbus0.metaName = Simple
com.connecterra.ale.plc.plcMessageConvention.simplemodbus0.plcTransport =
modbus0

#Define inboundMessages 'msg1' and 'msg2', which are connected to PLC input
# IO 5 and 6 ('d' is a convention referring to discrete input in ModBus
```

```
#protocol)
com.connecterra.ale.plc.inboundMessage.msg1.plcMessageConvention =
simplemodbus0
com.connecterra.ale.plc.inboundMessage.msg1.receiveItem = d5
com.connecterra.ale.plc.inboundMessage.msg1.matchValues = true
com.connecterra.ale.plc.inboundMessage.msg2.plcMessageConvention =
simplemodbus0
com.connecterra.ale.plc.inboundMessage.msg2.receiveItem = d6
com.connecterra.ale.plc.inboundMessage.msg2.matchValues = true

#Define the input trigger named 'plcmessage' using the PLCTriggerDriver
com.connecterra.ale.triggerDriver.plcmessage.class=com.connecterra.ale.trigger
types.PLCTriggerDriver

#Define the directional trigger 'door2' - timeout = 10000ms
com.connecterra.ale.triggerDriver.door2.class=com.connecterra.ale.triggertypes
.DirectionTriggerDriver
com.connecterra.ale.triggerDriver.door2.timeout=10000

#input trigger at point 'a'
com.connecterra.ale.triggerDriver.door2.aTrigger=plcmessage:msg1

#input trigger at point 'b'
com.connecterra.ale.triggerDriver.door2.bTrigger=plcmessage:msg2
```

Index

T

Z