The Seven Secrets of RFID Software Success

Introduction

RFID software or "middleware" has been touted as the next big category in enterprise software. Over the past seven years, Quake Global has worked with more RFID software vendors and developed more RFID solutions than any other company – well over at least a dozen. The only thing that is clear in the RFID software market is that RFID middleware is not destined to be the "next big thing". This may not be good news for many of the software vendors that are selling overly complex and expensive products, but it is great news for the RFID industry because adoption is rapidly becoming cheaper and easier.

What is RFID Software and what should it be doing?

RFID software, at its core, is intended to bring RFID devices to the edge of your existing network. There are some critical functions that RFID software must be capable of doing.

RFID software must:

- Filter raw data coming from the readers into a smaller set of events
- Control a wide variety of RFID readers, printers, and handhelds
- Translate raw data from the readers into a usable format
- Handle network outages or inconsistencies
- Commission (create and encode) EPC numbers

Outside of these core features, all of the other functionality in an RFID system can be handled using traditional EAI (enterprise application integration), workflow or application server products. It makes no sense to push complex user interfaces and workflow to a new platform that has less overall adoption and industry acceptance. Many enterprises already own an EAI system with workflow functionality. Unfortunately, many of the RFID software vendors have attempted to accomplish just that – recreate an entire EAI layer. It is important to look at what functions in the workflow are driven by pure RFID data collection and need a new software base to support them. It is also important to see which functions can be fulfilled by existing systems or standards-based software.

RFID software is relatively new to market; therefore the amount of variance, in features from one vendor to another, is tremendous. Some vendors support only basic filtering of events in and out of the read areai, whereas, others allow complex filters that allow you to look for specific tags or even classes of tags. In addition, the way EPC commissioning (programming a specific number, in this case the electronic product code or EPC, on the tag) is handled varies significantly. Some vendors rely on ancillary software to generate the EPC number, while others provide a comprehensive and distributed way of assigning EPCs.

In terms of RFID reader and printer support, looking through the lens of a maturity model will yield the best results. The more mature vendors provide some level of device support for nearly every major reader and printer on the market. The less mature players provide support for a limited number of reader and printer vendors and their support is quite limited. However, most marketing material will claim support for all devices. It is important to note that even some of the more mature players are limited in the amount of control that they allow you for any given reader. This may mean that you are not able to control key variables such as antenna power or antenna cycling. This is particularly true of companies that do not have an RFID laboratory or a history of Benchmarking equipment, since they don't know all the granular information around optimizing readers.

Software Vendor Categories

When speaking to the different vendors, it is important to understand that each vendor is approaching the RFID market from a different angle, and their approach drives their product development as well as their commitment to the market. The vendor categories are:

- Enterprise application vendors
- Infrastructure software providers
- RFID pure plays.

When it comes time to look at the vendors, always keep in mind their end goal. Are they looking to sell software that pushes a new workflow tool? Perhaps they are promoting software that accomplishes a specific RFID task or are they pushing an ERP or WMS platform.

Enterprise Application Vendors

The enterprise application vendors such as SAP may be a great choice for you if you already use their WMS or ERP package, but among the worst choices if you use one of their competitors' software packages. The long-term strategy of application vendors is generally to get you to adopt their broader stack. These vendors shine in integration of things like product master data and GTINs with EPC commissioning. They tend to fall down as you try to vary the use cases that you are implementing. For example, batch verification or dock door reading may not be possible with many of these vendors. In many cases, these vendors will require you to add actual RFID software on top of the base in order to fill out the full stack. In addition, any RFID specific features are generally reprioritized. For example, most SAP implementations require RFID device control software in addition to their Automated ID Infrastructure package (AII). Red Prairie has also eliminated their RFID support.

Infrastructure Software Vendors

Software infrastructure providers such as IBM and Microsoft are often times a good choice when there is going to be extensive integration into existing systems, but a poor choice if you have to get simple RFID applications up and running. These players view RFID as another data collection mechanism that can be used with standard solutions. This view typically means that the core functions go unimplemented or they are poorly implemented. Some of the worst read rates are symptoms of an uneducated deployment using BizTalk. In depth device support is often lacking and any RFID specific task takes weeks to complete. These vendors generally are trying to sell services to implement workflow on their enterprise middleware platforms. The projects invariably become extremely large and complex, even if it is just a simple data collection exercise.

RFID Pure Plays

The final group is the RFID pure plays. The pure plays fall into two general camps: those who want to become a workflow provider and those that focus on RFID device control. Many of the players in the RFID industry started by creating small stand-alone applications that worked with one or two RFID readers. This is the case with companies like OATsystems and Xterprise. Over time, their platform grew and their sales pitch evolved. OAT and Xterprise started selling workflow engines that had RFID support rather than RFID software with some workflow. This has led to poor RFID device support over time and inadequate workflow support. It basically has become the worst of all worlds because no part of the stack is stable or mature, and instead bad workflow results. Any new use cases are expensive to build and difficult to maintain.

The RFID pure plays should be and are leading the way in some instances. The best architectures focus on the key device support and integration, and deemphasize the focus on workflow. For example, Quake Global's EasyEdge™ agents run on all of the leading handheld and fixed readers on the market. The agents control all of the reader interaction so that read accuracy is maximized, and reliable device control is assured. The agent simply forwards the reads using standard enterprise means of messaging like SOAP Web Services, JMS (Java Messaging Service) or database calls. This simplifies integration into existing systems and leaves the RFID focus squarely where it should be – on optimizing the read performance and creating a high availability network. Any workflows are easily handled by existing systems or a small new application can be written in any application server. This requires less work than using the other options because the application is receiving data like it would from any other enterprise source and the developer is not required to handle RFID events in a special way.

Seven Secrets of RFID Software Success

Now that the vendor landscape has been painted, let's look at the secrets that lead to project success in RFID software; in short there are seven:

1. Clear focus on the device
2. Alignment with Enterprise Architecture
3. Standards based integration
4. Eschewing of complex workflow in the RFID software
5. Moving from stand-alone to integrated
6. Built-in deployment best practices
7. Solution-centric: bringing together the best in hardware and software

These items are the key to getting RFID projects up and running accurately and quickly without excess development costs. Let's step through each one and delve into the keys behind each secret.

Clear focus on the device

Probably the most important secret of all is to have software that puts most of the development time and focus into the device itself. Many vendors have basic support for a device and simply issue basic read commands using a polling mode or an event mode. Your RFID software needs to know about all of the different parameters on the device and be capable of operating in multiple read modes on every device.

The Leader in Remote Asset Tracking

RFID software vendors often times ignore these tasks because they're time consuming and invisible, but the depth of the device support directly correlates to the accuracy of your overall RFID system. If you are trying to get close to 100% read rates, you will need to be able to use all of the features that the reader vendors are providing in order to maximize your read rate.

The software on the reader needs to manage all of the following, the blocking and tackling of RFID:

- Controlling when to start and stop reading (either with photo eyes, time based, or event based)
- Filtering the data so that reads are not duplicated
- Eliminating stray reads
- Grouping read messages so that reads are either sent in sets or individually
- Buffering data on the reader during network outages
- Sending the read messages onto a back-end system
- Controlling GPIO controls attached to the reader such as light stacks

These are the core features that are unique to RFID and are required to make an RFID system work. Any other workflow functionality in the system is ancillary.

Alignment with Enterprise Architecture

Outside RFID, almost all systems are message based. If you are integrating into an ERP system, the integrated system will send a message to the ERP system, and the ERP system may send messages back. This messaging may happen using a messaging framework like JMS or it may use web services or database access. RFID should not be the exceptional system where the ERP system or application server is attempting to manually control a device and poll for information. This causes a myriad of issues related to scalability, reliability, and read accuracy. Not using a message based architecture usually results in poor read rates and a high latency system.

If your application is attempting to control many servers remotely, there are two problems. One, the reader to application round trip may be too long to respond, so servers will be required in every facility adding more readers to the network and bleeding budget. In addition, as you add readers to the network, the application server must increase its resources to handle the readers.

Ideally, your system should have agents that reside on the readers and utilize the compute power and memory on the readers. This fixes the scaling and network latency issues. As readers are added to the network, the system scales with the compute power on the readers. Your application server infrastructure can remain centralized.

Those agents should then be capable of filtering the data and sending it in standards-based messages to your back end systems. This allows you to treat RFID from an application perspective the same way you would treat any other 3rd party system. Your overall enterprise strategy stays intact, and additional applications are easily added.

The Leader in Remote Asset Tracking

QUAKE
GLOBAL

Standards Based Integration

The integration in an agent-based software is easily handled if the messages are sent using standards based protocols like JDBC, JMS, or SOAP. There should be no need to have a middle system process the RFID reads and then send them onto the back end systems in another format. There are vendors that spend large amounts of time trying to get read data from their system, and a simple sending of the filtered tag reads will accomplish 90% of the required functions. The remaining 10% is solved with your existing systems.

Eschewing of complex workflow in the RFID Software

Complex workflows are the bailiwick of WMS systems, ERP systems, and enterprise workflow tools. The RFID systems should feed these systems, but should not attempt to create specific workflow. Existing tools offer scalability and ease of development, and if RFID adds read data to the mix, new and enhanced applications are easily developed and adapted. By implementing complex workflow in the RFID system, you will be endeavoring in a large development project that will invariably have problems because it is on a system not tested for scale.

Moving from stand-alone to integrated

Often times it is best to start with RFID in a stand-alone mode and then move to an integrated model. This is where many of the RFID vendors originally gained traction by offering a full system that does everything. A stand-alone system can be the best way to start, but it should be implemented in a way that the workflow can be removed over time and the integration can remain. This means developing simple workflow in rapid development tools and feeding the system with data like you would feed an enterprise system. This allows you to get up and running quickly and cheaply and then move to a scalable back end over time.

Built-in deployment best practices

Part of a device-centric approach is understanding how the devices are used, their strengths and limitations, and providing software that comes out-of-the-box ready to handle the most common RFID use cases. These use cases include supporting hardware configurations such as RFID tunnels, portals, and cabinets. Each of these scenarios comes with its own idiosyncrasies, such as determining the direction items are going for portals, utilization of multiplexors and antenna configuration placement for cabinets, and managing read triggers (such as photo eyes) to minimize stray reads for tunnels.

Solution-Centric: bringing together the best in hardware and software

RFID Readers do not exist in a vacuum. At Quake Global, within our 275+ projects, we have seen almost every type of deployment scenario and use-case there. This contributes to our hardware engineering. A complete RFID solution does not end at the reader, but requires bringing together a variety of hardware and software components to create a unique and dependable solution. Lone hackers in cubicles cannot design great software solutions: they need access to the right facilities and people to build truly production-ready, battle-tested solutions. This means having a world-class laboratory and having a team that spans several disciplines, including RFID, mechanical and industrial engineering. If your software vendor does not have all of that, they may have software, but they do not have a solution. This may not be obvious during the sales cycle, but it will become glaringly obvious once the implementation starts.

QUAKE
GLOBAL

Wrapping it Up

RFID software is often the most frustrating part of any deployment, but it doesn't have to be. It is one of the three legs that make the foundation of an RFID tripodii. Keep the unique RFID software to a minimum, find a partner who has a complete device solution and a background in physics so the devices work well, and then manage the system just how you would manage any other enterprise system. There is no need to buy large, complicated workflow engines, or endeavor in creating new device drivers. There are solutions out there that work with the major readers on the market and integrate easily into any back end system. In the end you don't need to rebuild SAP you only need SAP to work better through RFID.

QUAKE
GLOBAL