

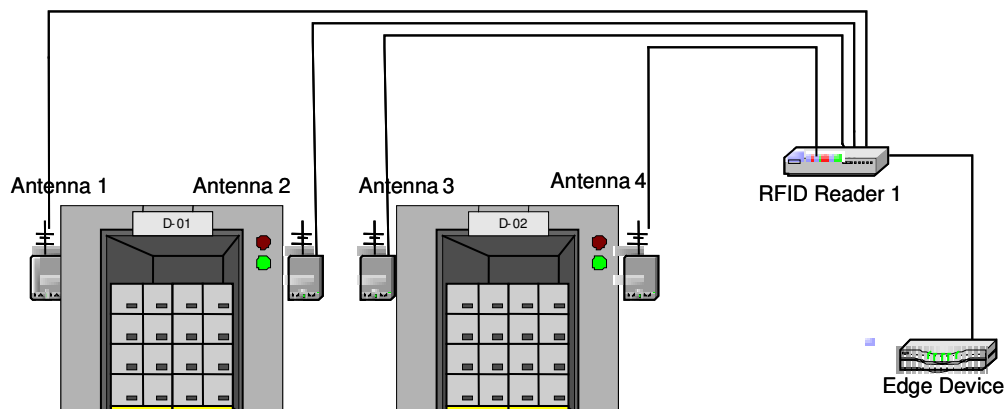


Should You Push or Pull RFID Data?

*By Mark Fralick
SupplyChainDigest Contributing Editor*

In my last RFID column, I explained the importance of understanding how you and/or your RFID vendors will deal with the “context” of read data. Here’s why: the processes occurring in the facility coupled with the integration (communication) ability of the execution system (such as a WMS) will ultimately determine where context decisions are made and, therefore, where the processing is done. I like to think of this as the “push or pull” decision (although there is a case for a combination of push and pull).

We’ll consider by using a typical distribution example: truck loading. We have a set of dock doors similar to those in the figure below. Each dock door could be equipped with photo-eyes under the gateway frame where the antennas are mounted. Additionally, a light stack may be used to visually indicate to the fork truck driver that pallet is correct and accounted for.



Truck Loading Scenario

Push Example

In this situation, pallet jacks are loading a truck and are not being controlled by a WMS. Therefore, the Edge, using RFID “middleware” functionality, is doing the bulk of the context work. In addition to the readers, the photo eye and stack light are controlled by the Edge Device. The Edge Device uses presence detection from the photo-eye to activate the reader(s) for the corresponding door. The edge interrogates the tags for a pallet tag and any case tags. The processing might look like this:

- ❑ We set rules up at the Edge to indicate we must see a pallet tag and some number of case tags. If not, the Edge sets the stack light to Red. No communication of data to other application systems is needed.
- ❑ The Edge might also have knowledge of the association between the pallet tag and the case tags. If so, this check can be done down at the edge. If something doesn't match, light goes to Red. Again, no communication to other applications needed.
- ❑ Assuming we've passed all the Edge-based rules, the Edge then pushes tag read data up to the other execution systems/applications. If the Edge had previous knowledge of the association between the pallet and case tags, it might also possess the association between the pallet tag and the pallet ID, which the warehouse system should know. If this were the case, all that is needed is to send the warehouse system the pallet ID and the dock door number (associated with the reader). Otherwise it might send pallet tag, potentially pallet ID (if the Edge is aware of it), case tags, potentially the GTIN numbers or SKUs of the cases, and the dock door (associated with the logical readers at the Edge level).
- ❑ The higher-level execution system then does its work, making sure this pallet should be going onto this truck, changing its location status, etc., and then passes back a completion status. Based on the completion status, the light stack display is set appropriately by the Edge software.

Pull Example

The pull example is driven by the execution system. In this situation, the pallet jacks are connected to a WMS via RF terminals. The operator performing the load operation scans a pallet in the staging area. At this point, the execution system directs the operator to the correct outbound truck. The execution system detects presence at one of the photo-eyes.

- ❑ Since the execution system has more knowledge of the logical context, the first check it might make would be to verify that one of the RF devices has picked up product going to this door. If there is a problem, it could immediately illuminate the red stack light.
- ❑ The execution system would now tell the Edge Device to sample the field and return all the unique tags it read.
- ❑ The execution system takes this list, which might include a device tag, and performs any needed validation checks.
- ❑ Based on the result of these checks, the operator would likely be informed via the stack light and the RF display that the operation was successful or had a problem.

Advantages of the Push Approach

- ❑ The push example off-loaded a lot of device-level interfacing (photo-eyes, stack lights, etc.) to the Edge. This is important if the execution system can't easily deal with these types of device.
- ❑ By permitting the Edge to do all of the low-level coordination, the only data moving above the Edge was clean, filtered data. In fact, by previously associating case IDs with tag IDs, or the tag IDs with the pallet ID, the Edge may have the ability to communicate very useful information instead of just tag data.

Advantages of the Pull Approach

- ❑ The execution system has better logical context. It knows, for example that it told RF unit 01 to move a certain pallet to a certain truck. This gives it some context for what is going on.
- ❑ The execution system can control the transaction end-to-end. Since it is controlling the entire transaction (from pickup to monitoring put-down) it has a better chance of incorporating status data from photo-eyes or data from the reader into the process flow.

Conclusion

The Edgware vendors (companies like Connectera, Globe Ranger, Oat Systems, and RF Code like to think they should be doing all the work having remotely to do with RFID and trigger devices. The execution vendors (e.g., WMS suppliers) are scurrying around saying "No, we can do that" (many of them actually using the same 3rd party Edgware software to manage the reader interface). Ignore the noise and focus on context. In general, push context processing down as far as possible. But understand that there are tradeoffs that might be beneficial to the execution system if you move some of the work up one domain. There is no rule that says the Edge must do this part of the work and the execution system does that part. Understanding your context decisions and the flow of your execution application will help you map out the responsibilities.

My next column, called 'Architecture, does it still matter?' will explore questions of architecture in the Supply Chain Execution space. Stay tuned...

Delivering ROI without the "blah, blah, blah"

About the Author

Mark Fralick is president of ROI Solutions and a SupplyChainDigest Contributing Editor. Having operational, implementation and software development experience, along with detailed understanding the workings of software and services companies, puts ROI Solutions in a unique position to stand strongly on the customer's side in the battle for ROI in system selection and implementation. Learn more at www.GetUsROI.com, or email Mark Fralick at Mark.Fralick@GetUsROI.com.

About SupplyChainDigest

SupplyChainDigest™ is the industry's premier interactive knowledge source, providing timely, relevant, in-context information. Reaching tens of thousands of supply chain and logistics decision-makers each week, our flagship publications - *SupplyChainDigest* and *SupplyChainDigest – Logistics Edition*, and web site (www.scdigest.com) deliver news, opinions and information to help end users improve supply chain processes and find technology solutions.

For more information, contact SupplyChainDigest at:

937-885-3253

www.scdigest.com

email: info@scdigest.com