

# ALE: A New Standard for Data Access

Application-Level Events, one of the first RFID software specifications EPCglobal is expected to ratify, promises great benefits.

Apr. 18, 2005—In October 2003, EPCglobal was created to commercialize the work of the MIT Auto-ID Center, through the development of Electronic Product Code (EPC) standards for the use of passive RFID in the supply chain and other sectors. While EPCglobal's work on the UHF Gen 2 hardware standard and the EPC Tag Data Specification are well known, EPCglobal has also been hard at work on the creation of software standards for use in RFID and EPC systems. One of the first such standards expected to be ratified by EPCglobal is called “Application-Level Events (ALE).”

Everybody who has tried to build an application using RFID readers discovers that RFID readers produce a flood of very low-level information. Essentially, a reader continuously scans for tags that are near its antennas, as often as three or four times per second, and each time reports every tag it sees. This raw stream is simply not an appropriate starting point for writing business logic. Business applications want to start with a more meaningful statement of “what, when and where.” For example, a “smart shelf” application wants to know about tags that are added or removed from the field of the reader, while a receiving application wants to know the total number of distinct tags that are observed at a door each time a motion detector is triggered.

The transformation of the raw tag read stream into “what, when and where” statements is generically referred to as “filtering and collecting,” and the need for it has been recognized since the earliest days of the Auto-ID Center. The researchers at the Auto-ID Center coined the term “Savant” to describe this function but made no attempt to standardize the specific kinds of filtering operations that would be available for applications. The question is, if each application needs a slightly different starting point, how can a standard be created?

The first answer to this question was provided by Procter & Gamble’s director of IT, Steve Rehling, who did a survey of more than 25 different RFID use cases. His conclusion was that while each use case had different business logic, the filtering and collection requirements boiled down to much smaller number of specific operations, including removal of duplicate reads, aggregation over intervals of time, grouping, counting, differencing and filtering (that is, removing certain EPCs based on the company ID, product ID or other data on the tag).

Out of this work grew Application-Level Events, which for the first time specified an interface through which an application could indicate exactly what information it wants from the raw stream of RFID reads. Through ALE, an application can specify a number of things:

- Which locations it wants to read from (where a location maps to one or more RFID readers or antennas)
- What interval of time to accumulate data (absolute time, triggered by external events such as motion detectors, etc.)
- How to filter the data (e.g., include only Acme products, exclude all item-level tags)
- How to group the results (e.g., by company, by product or each tag separately)
- Whether to report currently visible tags or just additions or deletions
- And whether to report actual EPCs or just a count of the tags that are read.

A request of this kind may be made in an on-demand mode, where the reads are performed in response to an application request, or as a standing request, where data is sent to the application periodically without further request.

If this paradigm seems familiar, it is: The ALE paradigm is very similar to the way SQL (structured query language) is used to provide applications access to database data. Through SQL, an application makes a high-level request for data—which rows and which columns, of which tables—and the database server determines the most efficient plan to fulfill that request. Likewise, an RFID application makes a high-level request for data through the ALE interface, and the hardware or software on the other side of the ALE interface fulfills the request. ALE plays the same role as SQL in shielding applications from a lot of low-level implementation detail. For example, if an ALE-based application asks that RFID reads be filtered to include only Acme Corp.'s products, the filtering might be done by middleware that interacts with a reader, or it might be pushed onto the reader itself, depending on the capabilities of the reader hardware and the middleware software.

Another key benefit for end users is that ALE facilitates sharing RFID data among many applications simultaneously. Different applications may make requests through the ALE interface, without requiring those applications to have knowledge of each other. If two applications request data from the same reader, the implementation of ALE mediates those requests and makes sure that each application receives the data it needs.

Here is an illustration of how the data-sharing aspects of ALE allow an end-user company to get more value from its own network of RFID readers. Imagine you are a large global retailer. In your distribution centers, you have readers that are talking to a warehouse management system (WMS) application. You have readers in the front rooms of your stores talking to a point-of-sale application. You have readers in the back rooms talking to an inventory application. In all, you have perhaps a dozen different enterprise applications using RFID data. So far, so good. Then one day, you receive an urgent e-mail from a pharma supplier, saying, “There's a bad batch of aspirin; here are the EPCs for the bad cases; please find them and destroy them.” Your existing enterprise applications probably have no specific search-and-destroy feature.

But if you've deployed an ALE-based infrastructure, you can easily make a simultaneous request to use your entire RFID network to do reads and find the EPCs for the bad aspirin, all without disturbing the existing applications. Through ALE, you're able to exploit your investment in RFID readers as a multipurpose facility for locating tagged objects, rather than viewing each reader as a peripheral for a specific application.

ALE truly represents a significant advance in the way RFID infrastructure and applications are built. More than 60 companies participated in the EPCglobal working group that created the specification. Last November, six companies created implementations based on the ALE specification and brought them together for two days of interoperability testing—more than 30 combinations were tested in all. My company, ConnecTerra, has had an ALE-based middleware product on the market for more than a year. Globe Ranger released an ALE implementation last September. With the specification nearing ratification, other companies are likely to follow suit.

ALE is likely to be the first software standard to be ratified by EPCglobal, and it promises great benefits to end users, including reduced time to create applications and a wider array of interoperable technology choices. Other software standards are expected to be ratified later this year, so stay tuned!

*Ken Traub is CTO of ConnecTerra. He is cochair of the EPCglobal Filtering and Collection Working Group creating the ALE standard, and is the original author of the ALE paper along with Steve Rehling of Procter & Gamble and Ted Osincki of the Uniform Code Council. He is also a member of the EPCglobal Architecture*

*Review Committee, which oversees all EPCglobal technical specifications.*

Copyright ©2005 RFID Journal, Inc. All Rights Reserved