

# Inside Analysis article – Events that Change the World

I don't know whether you have noticed, but we have been slowly moving from a transaction-based IT world to an event-based IT world. Think of it like this and it becomes clear. We used to focus on transactions, things that directly impacted the business. Now we care about simple events like a single log record or someone clicking on a web link or someone writing a tweet. We could argue that we have acquired the computer power to process a finer granularity of data. And that's true. If you analyze any "old-world transaction" you quickly discover that it consists of a bundle or cascade of events that we used to treat as one thing.

## Think About Dimensions

But there's more to it than that, and some of it is fundamental. For example, the dimension of time is fundamental. In the past, you may well have built systems that dutifully time-stamped every transaction in the database. It was, after all, a best practice. Nevertheless we didn't care about time in the way that we do in an event-driven world. One reason we didn't care so much was that we lacked another important dimension: location. When you have time and place together, you have something very useful.

We began to care about geography as soon as smart phones with their mobile applications proliferated. It wasn't just that you could know where someone was with reasonable certainty; it was also the fact that a restless network of handheld devices was spread out across almost every town and village on the planet. News began to happen via Twitter, and the reporters were whomever was nearby that had a Twitter account. Uber, the virtual taxi company, simply leveraged geography. Nowadays, many mobile apps care about geography and would have less

value if they didn't know about location.

Let's examine the reality of an event.

Events have a time and place. Think about this in data terms and you realize that while in the previous transaction-oriented way of looking at the world, a data record might have a single fairly simple identifier (i.e., key). Rather than having simple keys, events have multiple fundamental indexes. Events always have a time and place. And place, by the way, can be simple, like a map reference (latitude and longitude) or more complex – since a map reference may identify a building. But for some events you may need to know exactly where in a building an event originated. So time and place might be as complex as: time (date and time) and geography (map reference plus 3D location within a building).

There is also the source of the event, in the sense that the data may be created by something as simple as a sensor or as complex as an application running on a server. There is also likely to be an event type such as a sensor report, log record, app message, SMS message or phone call. Now if the source is mobile and has a person attached to it, the context of that person will be important. For some events, this may mean that you need to know or capture someone's event-associated social network: family, friends, co-workers and so on.

Personally, I tend to think of the Internet of Things (IoT) as made up of "plants" and "animals." The plants don't move. They stay where they are reporting data when there is something worth reporting, but the animals (mobile phones, cars, trains, airplanes, etc. and their associated people) move around and thus can change their context considerably.

## **So, What About a Platform?**

The landscape of event data I just described would be entirely hypothetical if there were no software that could capture and

process the event data as described. In fact, there is. Of course you can capture event data, even multidimensional event data, in any data store. The real need is to store and manage the data in an intelligent way and to build applications on top of it.

This is where AllegroGraph (from Franz Inc.) shines. It is an RDF Graph database – although in my view, it is best thought of as a platform that is particularly suited for building apps that process event data. Because it is a Graph database, it can store pretty much any kind of data and query it, not just in the time-worn relational fashion, but also in a graphical manner – carving out graphical maps of relationships. And on top of that it can apply semantics to deduce as-yet-undiscovered knowledge from the data. As far as I can tell, AllegroGraph goes much further than other products of its ilk, providing a very rich and well-thought-out set of indexes to the data it stores.

It is far too sophisticated a product to describe in a simple blog post, so rather than discuss its technical implementation or its user interface, I'll simply list some of the general kinds of questions it can answer, beyond the usual database capability. Here's a selection:

- In what order did a specific set of related events happen? (Temporal query)
- Are there patterns of events in our data that seem to be related by time? (i.e., Are there temporal data patterns we have not yet recognized?)
- How far apart in a (social or physical) network are two “actors” and how strong is their relationship?
- What are the identifiable social groups and what are the general patterns of such groups? (This applies to a specific entity, such as customer, staff, citizen or soccer fan).
- How important is any given “actor” in any given network and event?

- What messages of a specific type emanate from a specific area? (Geographical question)
- What happened, when and where?

This last question could be further generalized into the what-when-where-who-how-and-why question – which is a good way of summarizing the potential of this technology. With semantic techniques you can certainly get at “how” and you may be able to test hypotheses about “why.” AllegroGraph can do that and its indexing will give you the rest, either individually or in combination.

In my view, AllegroGraph is a product worth taking a good look at. Its capabilities are very broad, and they provide a glimpse of the shape of things to come.