The Atlantic Systems Guild Ltd.

What's the Point of Business Events?

Whatever it is you are working on, it is almost certainly made up of many, sometimes very many, pieces. Each of the pieces interacts with other pieces to achieve some valuable result. This means that there is a need to organise all the pieces so that you and your colleagues can work on relevant slices of the larger problem, and at the same time keep track of how the smaller pieces work together within each slice. The Business Event is the best tool we have been able to find to help you do this.

By Suzanne Robertson & James Robertson – the Atlantic Systems Guild

What is a Business Event?

A business event is something that happens, and when it happens it causes a pre-planned response by the business, or as we shall call it here, "the work". One category of business events are the things that happen inside an adjacent system. The work is made aware that the business event has happened because each happening produces a flow of data to the work. A business event is a significant happening – it is not just a mouse click. It is often a request for a service that your business provides, and the outcome is the provision of the service or product.

For example, Figure 1 illustrates the business event *Customer decides to pay a bill.* The adjacent system, in this case the *Customer*, is where the event happens. The data flow (caused by the event) is the *Customer Payment.* This flow is usually called the "triggering data flow" because it triggers a response by the work. In this case, the response would be to accept the payment, record it, and send a confirmation receipt. All this would be done according to the work's business rules. Note that these rules might be carried out by people, technology or a combination of the two.

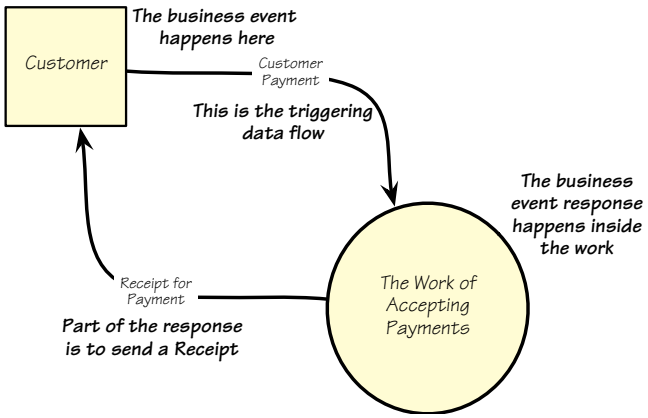

Figure 1: Summarises the business event and response.

Keeping Track of the Business Events

Whenever a new business event is confirmed to be relevant to the scope of the work, the business analyst keeps track by adding the event to the work context diagram. Figure 1 showed a work context diagram that only includes one Business Event: *Customer decides to pay a bill*. In Figure 2 there is an additional business event, *Customer makes an order*. You can see the additional flows for this event on the diagram.

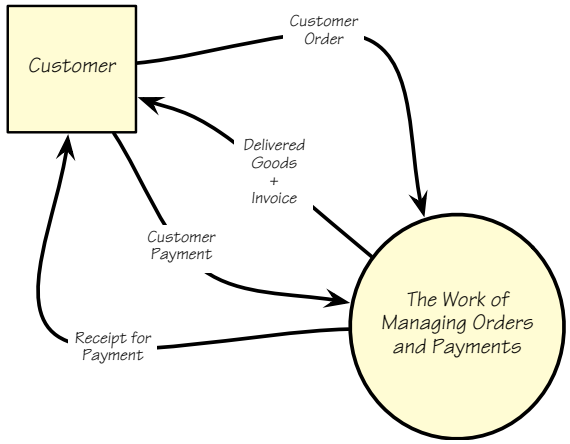

Figure 2: This work context diagram contains two Business Events.

Every time you discover a new business event, and it is agreed to be within the scope of the work that you are studying, it should be added to the work context diagram. Each business event represents a bounded chunk of functionality that you can study independently of the other events while keeping track of how the individual events fit into the whole. As the number of business events increases an event list (illustrated in Figure 3) helps you to manage and prioritise your investigation. Naturally enough, your list of events would be considerably longer.

Event Number	Event Name	Input	Output/s
1	Customer decides to make a payment	Customer Payment	Receipt for Payment
2	Customer makes an order	Customer Order	Delivered Goods + Invoice

Figure 3: The Business Event list is a convenient way to manage a large number of events.

A business event represents a significant and independent chunk of functionality that can be prioritised so that you are always working on the events that yield the greatest value. Business events also help to organise your investigation. You can apportion the events among your team, and due to the independence of the events, there is not a lot of need for team members to have excessive interaction.

Working with an agile development team

Business events are highly effective for agile teams. A business event (or correctly, the response to the business event), is a stand-alone chunk of the problem with a well-defined outcome. It also has well-defined input and outputs. This makes it not only a convenient unit to study and find its requirements, but also a sensible unit to develop.

Figure 4 illustrates this approach. As we have seen, the scope is determined by the collection of business events, and for each of these, we advocate writing a *business event story*. This is a story in the conventional sense of role-function-outcome. More on this later.

For each business event story the developers, with the help of the business analyst, can define the details of the response to the business event by writing a number of *functional stories*. These represent a breakdown of the functionality. They can be further broken down by the developers by writing the *detailed tasks* necessary to implement the functional story.

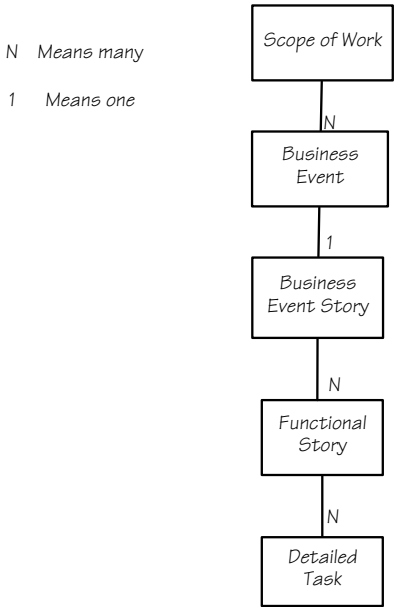

Figure 4: The requirements hierarchy provides the basis for iteratively working on the business requirements and the implementation requirements. The scope of work contains many business events, each business event has one business event story. Each business event story has many functional stories. Each functional story has a number of detailed tasks needed to implement it. This hierarchy is effectively managed using a story map.

Figure 5 suggests a traceable technique for deriving the Business Event Story for a Business Event.

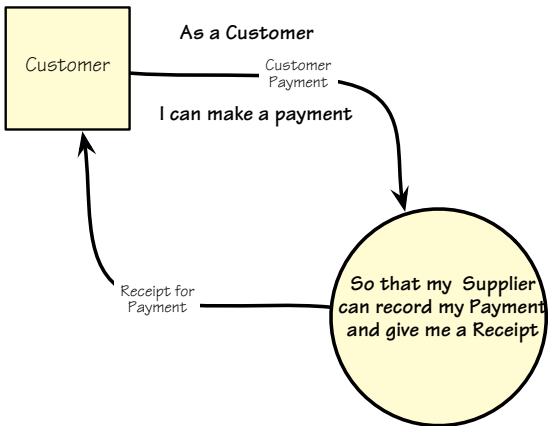

Figure 5: The Business Event, the triggering data flow and the event response are the components for building the Business Story.

The resulting Business Story is:

> As a Customer
>
> I can make a Payment
>
> So that my Supplier can record my Payment and give me a Receipt

With this approach, the developers can derive the Functional Stories from the Business Story. In this example the Functional Stories for this Business Story would be something like *Locate Customer Account, Record Payment, Issue Receipt.*

If there are changes or questions to either the Business Story or the Functional Stories, business analysts, product owners and developers have a common communication language.

Other sources of information about subjects discussed in the article:

Business Analysis Agility: Delivering Value Not Just Software
https://amzn.to/2CnVhTN Video https://youtu.be/l_9u5UI12uo

Reviews on many books by authors who continue to enrich the field of requirements and business analysis at https://www.volere.org/resources/books/

Or on the Volere Requirements LinkedIn group
http://www.linkedin.com/e/vgh/2491512/

Suzanne Robertson and James Robertson are principals and founders of The Atlantic Systems Guild http://www.systemsguild.com and joint originators of the *Volere* requirements process, template, checklists and techniques https://www.volere.co.uk

You can contact the authors at

suzanne@systemsguild.net

james@systemsguild.net