Continuous Design

Incremental and iterative design with Continuous Design.
By Bart de Best

Context:
At a Small and Medium Enterprise (SME) organisation there was a need for control over software development. The management of the DevOps teams that had been outsourced was completely lacking. The assignments were given via collaboration tools and realised immediately without a design or a product backlog tool such as Jira. This gave the business the feeling of switching quickly and steering directly. Over time, however, it became apparent that many items had not been delivered, deliveries were becoming increasingly slower and what had been delivered did not meet expectations. So it's time to introduce some control into the development process, but then Lean and Mean.

This blog is in line with the blog 'Incremental and iterative planning with Continuous Planning' but is written in such a way that the blogs can be read independently.

Challenge:
The challenge with this organisation was that the business was immediately in the lead without any knowledge of management such as planning, prioritisation, acceptance, and the like. The introduction of any form of control had to be justified, no matter how Lean and Mean. In addition, in this flat organisation, no one owned anything. This meant that any decision-making required a Polish country day with the entire (quite large MT).

Solution:
The solution to this challenge has been found in the concept of Continuous Design. This blog discusses this approach through the following steps:

1. Definition system context diagram
2. Mapping the value streams
3. Analysis of the value stream using a value stream canvas model
4. Translating the improvements into themes and epics
5. Detailing the design in a use case diagram and use cases
6. Completion through the use of building blocks

1. Definition system context diagram

First, an overview was made of the information provision using a system context diagram, as shown in Figure 1. The information system is shown in the middle. An information system is defined as an application including the required infrastructural facilities, plus the value streams that describe activities that take place automatically and manually and the users who use this information system to carry out the activities.

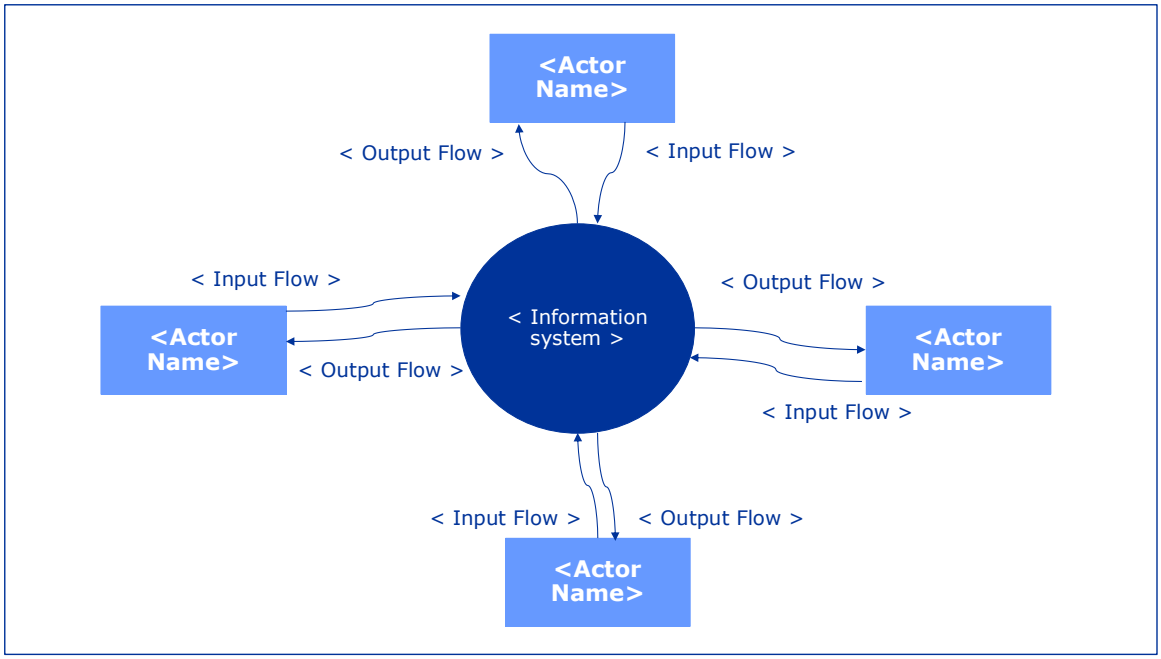


Figure 1, System context diagram template.

The actors are the stakeholders who interact with the information system. In this case, these are management who define financial goals and receive reports back. In addition, the end users who purchase services. Adjacent information systems are also actors that exchange information with the information system. In this way, all input and output of the information system is mapped. In itself this was an excellent way to quickly and clearly map out the scope of the design. Drawing up this diagram took one hour of workshop with three subject matter experts in which the diagram was drawn and approved. The drawing has been saved and described in Confluence.

2. Mapping the value streams

The system context diagram has been used to map the value streams supported by the information system. A total of 12 value streams have been recognised, as shown in Figure 2. A value stream consists of 10 to 15 steps that are called use cases. The completeness of the value streams was checked by mapping the input and output of the information system from the system context diagram to the value streams and inversely proportional. As a result, the system context diagram had to be updated because new input and output streams emerged when drawing up the value streams.

Each value stream was drawn and reviewed in a one-hour workshop. It has also been determined who owns the value stream. An exception of an important value stream where the workshop approach did not work. The manager involved was unable to provide an overview of the activities because they always seemed to be different.

It also turned out that work was being carried out that was not captured in the information system but was administered on a whiteboard. After recording a 4-hour interview, the analysis of the work was performed, completed, and reviewed. Only one adjustment was needed, and the manager involved indicated that she had never

seen such a clear flow of her work in the past 5 years. All value stream drawings are stored and described in Confluence.

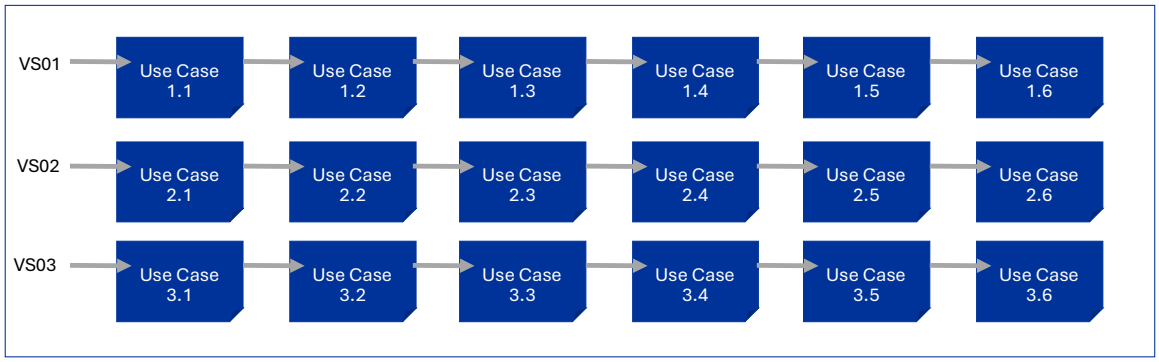


Figure 2, Value stream template.

Based on the insight gained into the administrative organisation, all outstanding activities have been mapped to the value stream for which they were carried out. This gave every value stream owner an overview of what was going on. The MT received a complete overview of all activities. These activities were then administered in Jira and linked to a theme. Figure 3 shows the link between the Continuous Design artefacts (green) and those of Continuous Planning (blue).

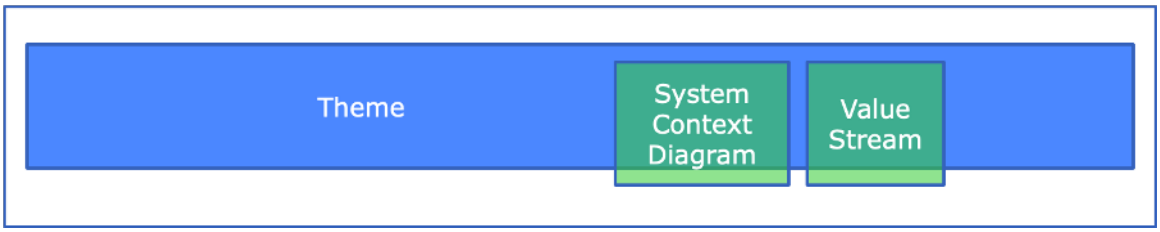


Figure 3, Continuous design linked to Continuous Planning.

This rudimentary control provided a lot of energy but was not enough to achieve universal priority. Parallel to steps 1 and 2, a business analysis was therefore carried out from Continuous Planning into the mission, vision, business goal and strategy of the organisation. The themes have been supplemented and refined based on this. This is described in the blog “Incremental and iterative planning with Continuous Planning”.

3. Analysis of the value stream using a value stream canvas model

After determining the themes and value streams, a value stream canvas was drawn up per value stream because the most important bottleneck per value stream can be determined. A bottleneck lies in the performance (limitation) or functional limitation (boundary). This has resulted in points for improvement for the value streams. The template used per value stream is shown in Figure 4.

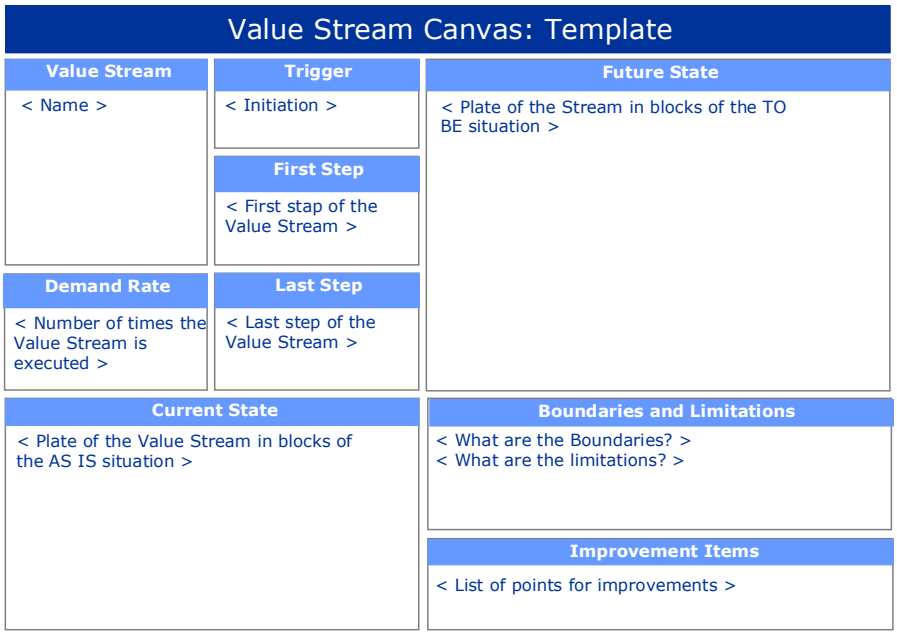


Figure 4. Value stream canvas template.

This template consists of those parts, namely the metadata of the value stream such as name, initiation, first and last step. The second part is the value stream as it is recognised and the bottlenecks in this value stream. The third and final part is the desired situation after the bottlenecks have been removed.

Each value stream mapping was completed in a one-hour workshop and documented in Confluence. The most important step was to estimate the lead time (Lead Time = LT), processing time (Processing Time = PT) and quality (Completeness and Accuracy = %C/A). This is also called Value Stream Mapping. The LT, PT and %C/A are called the Lean indicators. Initially it was usually indicated that this varies and that no figures are known. But by looking back at the past month and first assessing the end-to-end Lean indicators, this was quite possible. The delta size step between LT and PT was further examined for activity that could explain this difference.

This was often a waiting time that could be shortened by digitising tasks or adjusting the administrative organisation.
The PT times that were the greatest were also examined for waste and of course also the lowest %C/A. In this way, many issues were found in the value streams that people knew more or less about but had not quantified and therefore had not been addressed.

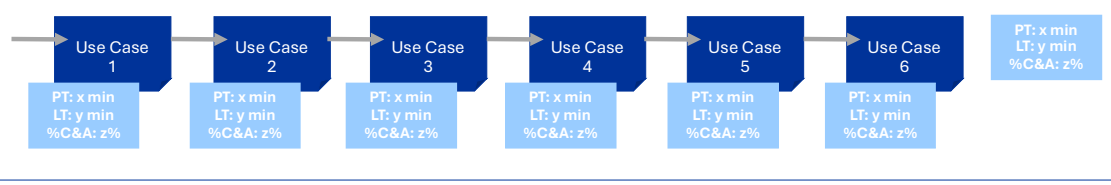


Figure 5. Value stream / Lean indicator mapping template.

A special situation was a team where the bottleneck cost 10% of the number of FTEs. This involved extracting spreadsheets from e-mail messages that contained information that had to be read into the information system. This bottleneck was removed within two weeks, allowing the saved 10% of that team's time to be spent on other important tasks that they never got around to. The value stream canvas models are documented and described on Confluence.

4. Translating the improvements into themes and epics

The improvements have been added to Jira in the form of themes, epics, or features. A theme is an improvement that equals the size of your Agile project, and an epic is a planning object of three months and a feature six weeks. Where possible, the planning objects have been merged with the themes from step 2.

An Epic one pager is described for each epic and includes the following information:

- Epic name
- Epic owner
- Epic goal
- Epic target group
- Epic functions (features)
- Epic hypothesis
- Epic risks
- Epic countermeasures

Also read the blog “Incremental and iterative planning with Continuous Planning”. The outstanding work (see step 2) that had already been mapped to value streams have either been added to the newly defined planning objects for the improvements or have been provided with new epics and features. In this way, a total product backlog has been created. Planning objects smaller than a feature are included as stories.

5. Detailing the design in a use case diagram and use cases

The value streams are detailed by graphical representations in which the actors and information objects are added to the value stream steps. This was slightly more interpretable for the business because the roles are expressed here.

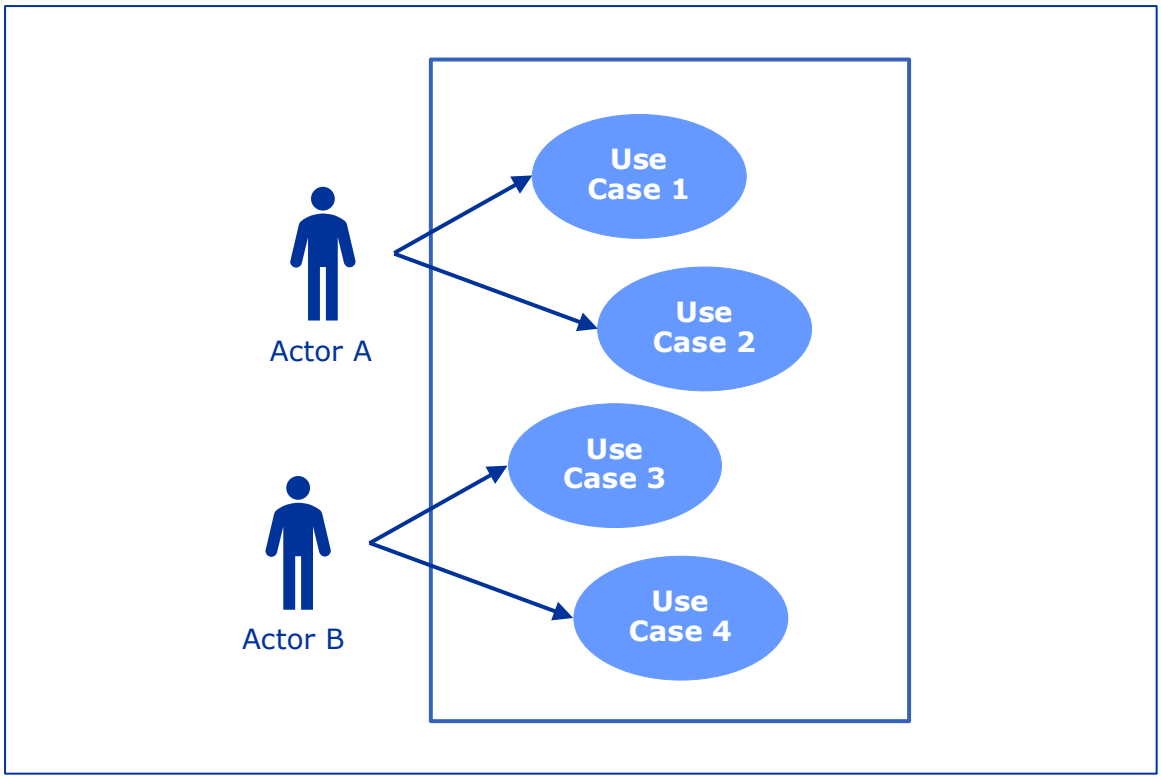


Figure 6. Value stream diagram template.

Each use case is described in a standard format. The following information is hereby documented:

- ID
- Name
- Goal
- Resume
- Precondition
- Postcondition
- Performance
- Frequency
- Actors
- Trigger
- Scenario with steps per actor
- Alternative scenarios
- Parent use case
- Interface
- Relationship such as building block ID

6. Completion through the use of building blocks

An application building block sheet was then drawn up in which the building blocks of the application are described. This plate consists of layers that are generic for each application. The building blocks themselves are unique. The template is shown in figure 7. It is then determined which value stream step uses which application building block as shown in figure 8.

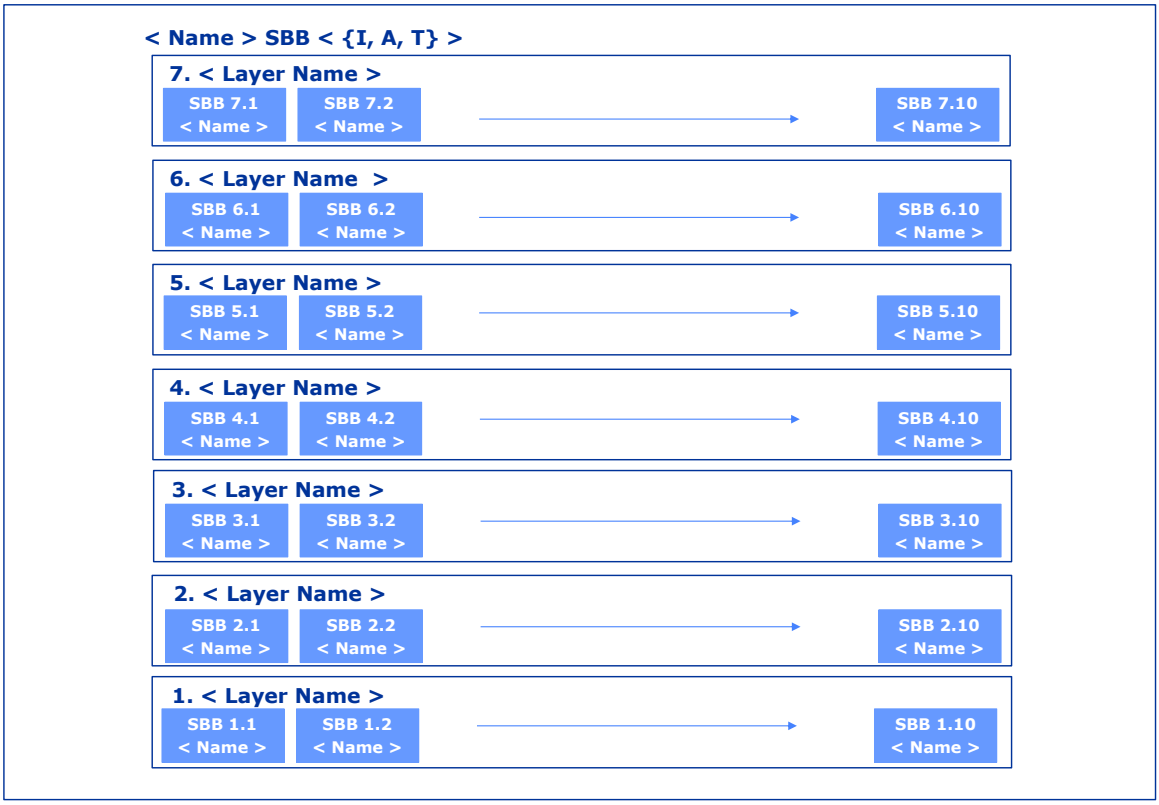


Figure 7. Application building blocks template.

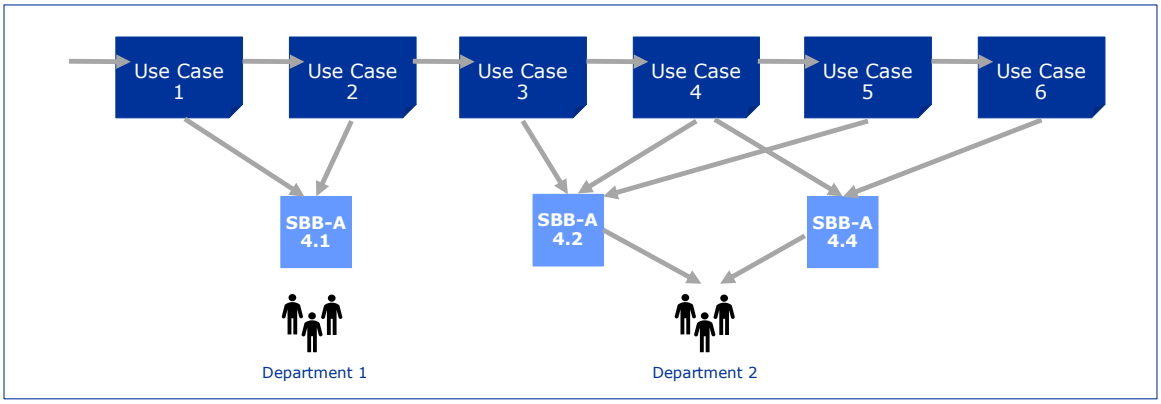


Figure 8. Value stream / application mapping template.

Based on the application building block plate, the completeness of the value streams could be checked, as well as the building block plate itself.

By mapping the bottlenecks based on the value stream canvas analysis, the epic one pager, the use case diagram, the use cases and the building block plate, the relationship between Continuous Design and Continuous Planning has been further clarified as shown in Figure 9. has been displayed.

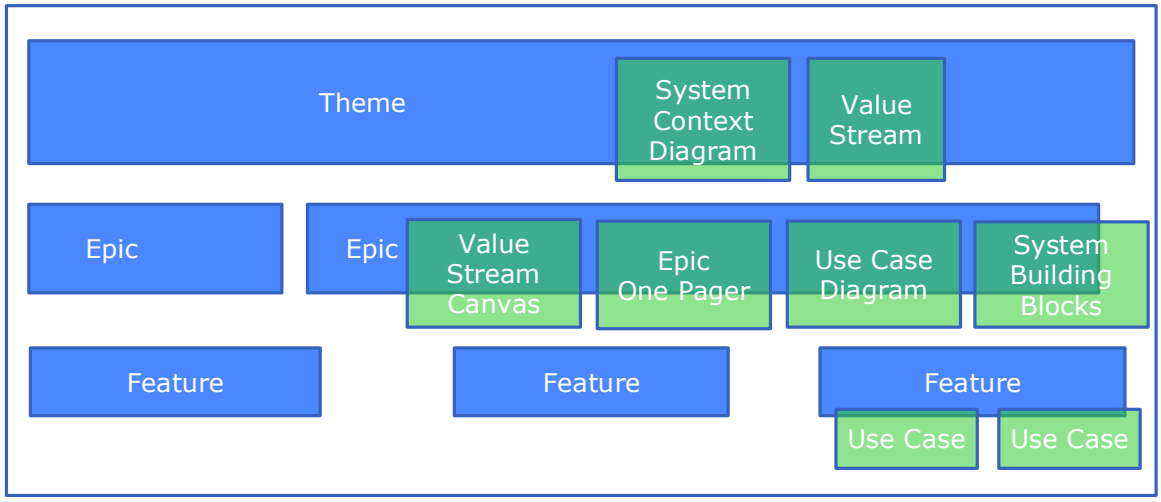
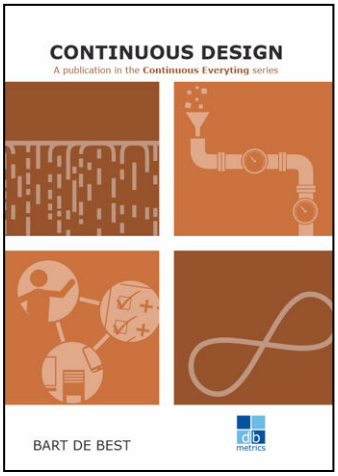


Figure 9. Coherence of Continuous Design and Continuous Planning.

Because the planning objects and design objects are consistently mapped, there is insight into what innovation and maintenance are taking place. This allowed the organisation to provide guidance by setting priorities. Partly by relating the business goal and strategy to the themes and epics with Continuous Planning, the feeling of being in control was created.

Because the product backlog is refined per sprint by the DevOps teams, the design is also refined and therefore grows along with the planning details. That is why this is a good example of emerging design shaped by the application of Continuous Design.

By Bart de Best
DutchNordic.Group



https://www.dbmetrics.nl/ce-en/continuous-design-en/