

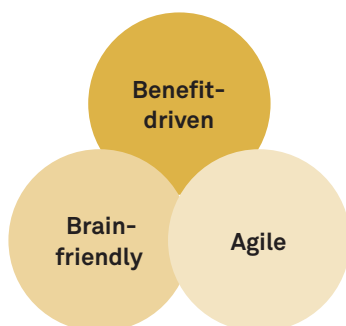
FOCUS ON BENEFITS IN AGILE TRANSFORMATIONS

– a Brain-Friendly Approach

implementconsultinggroup.com

By Christian Mark Christensen
cmch@implement.dk
and Rasmus Rytter
rry@implement.dk

Figure 1
Key elements when
implementing Agile in
large organisations



Transforming an organisation from working with waterfall models to working with Agile methods and mindsets is a large change for everyone involved.

Often, Agile starts in one corner of the organisation and is scaled from there. This approach can be very effective in the beginning and can create a lot of energy. But often, implementations like this lose their bearings and suffer a series of local setbacks. Agile becomes a word instead of the new way of working and thinking, and the benefits are never realised to the full extent.

Three key elements of Agile change

In this article, we will present three operational tools that can help you drive an Agile transformation forward in the

whole organisation without having too many local setbacks.

- First, transformations need to be **benefit-driven**, and benefit ownership must be clear.
- Second, we must make the change **“brain-friendly”**. By this we mean making the things we practice being good at easy, fun and rewarding for everyone involved.
- Third, Agile transformations must use Agile tools to create flow and transparency. We call that using an **Agile approach** to an Agile transformation.

Design the Agile transformation to realise benefits

When we are benefit-driven from the beginning in an Agile transformation, the kind of benefits we pursue in the change become very clear. Focussing on benefits will often be what makes managers across the organisation take ownership of the Agile transformation, thus creating a demand for the project throughout the whole organisation.

Working with benefits

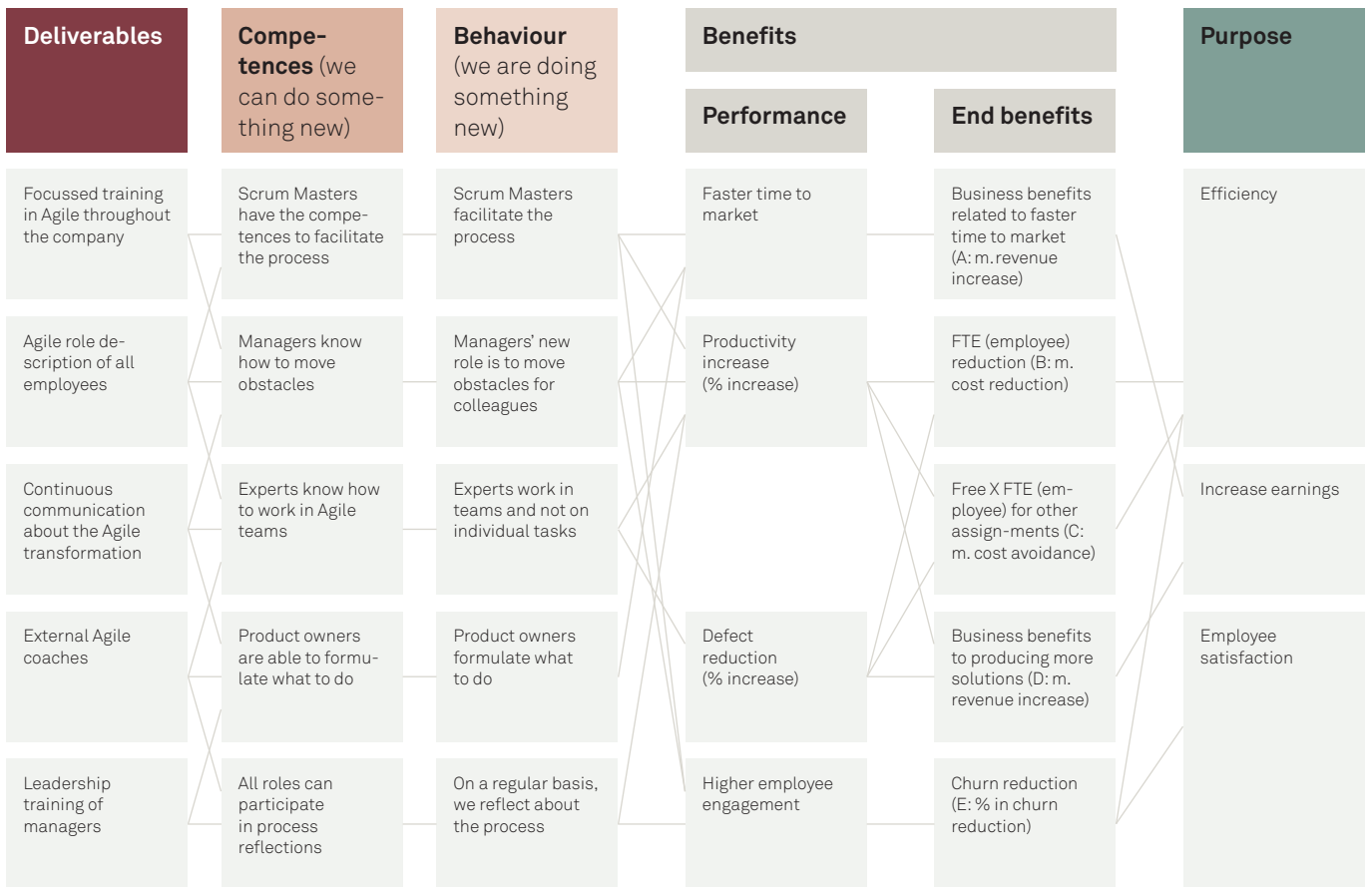
To ensure benefits realisation, we start by designing the Agile change at a three-hour workshop where the key stakeholders involved in the change create the benefit

map. At the workshop, we start with the purpose and work our way from the right side to the left side of the map.

The big difference compared with a more traditional approach is that we look at benefits realisation first and the specific deliverables last. Subsequently, the questions we ask are:

- What is the purpose?
- What benefits do we need to realise to fulfil our purpose?
- What changes will that require in our organisation?
- What change effort, training and deliverables or solutions do our colleagues need in order to change their behaviour?

Figure 2
The benefit map of an Agile transformation (simplified example)



Can we agree on the purpose?

The first step is to agree on the purpose, and the purpose is never to implement Agile! The purpose concerns the “why” behind implementing Agile. It could be “efficient solution delivery”, “enabling increased earnings” or “higher employee satisfaction”. If management in both IT and the business do not agree on the purpose, you will fail with parts of the implementation, and you are very likely to lose support from key stakeholders, which might jeopardise the whole initiative.

What are the actual benefits?

How big of an increase in productivity can we expect? Will the business promise to deliver more sales based on more output, and how much? Perhaps, we should instead use fewer defects and higher productivity to reduce the number of people in our organisation?

Making an organisation Agile can be tough and thus expensive, so if you don't have business or IT managers ready to own money-related benefits, it is not the right time to implement Agile.

Later in this article, we will address how we can ease the transformation with an Agile approach and a Brain-Friendly Approach.

Although we can make the transformation easier, it will still be a change in behaviour and competences that will demand an investment in time and resources.

What is our new behaviour?

The benefit map also provides an overview of the parts of the organisation that will be affected by the Agile transformation. Organisations working with Agile know that a large change in behaviour is involved when shifting from a waterfall approach to Agile methods and mindsets.

In the benefit map, we want to know what kind of new behaviour it will take to drive benefits realisation.

Examples of new behaviour are:

- Experts working in teams and not on individual tasks
- Managers who move obstacles for colleagues and who are not to be reported to
- Scrum Masters facilitating the process
- Product owners formulating what to do
- Teams formulating the “how to”
- All of us reflecting on the process on a regular basis
- Teams delivering working software frequently

By identifying the new behaviour, we will also have an initial idea of how big the change will be, and in order to actually experience the new behaviour, we need to support the change and develop new competences, which is the next step in the benefit map workshop.

New competences – when we want to do something new

Focussing on what kind of competences we need to have in order to acquire new behaviour makes it easier to identify the kind of training that is required. It may appear redundant that we distinguish between a Scrum Master who facilitates the process and a Scrum Master who has been trained and *can* facilitate the process. The reason is that going from *can* to *do* is the most difficult step in the benefit map.

The benefit map, which we have now unfolded, shows not only what kind of benefits we want to realise, but also what kind of behaviour and competences we need to develop in order to realise them. Using this approach makes it easier to choose among the large range of actual deliverables.

Deliverables to the Agile transformation

Our deliverables in an Agile transformation can be many, and it is very easy to lose our bearings. That is why we suggest using an Agile approach to an Agile transformation. We will return to this later in the article. Concrete examples of overall deliverables are:

- Agile role descriptions of all employees (technical deliverable)
- Focussed training in Agile throughout the company (training deliverable)
- Leadership training of managers (training deliverable)
- Continuous communication about the Agile transformation (change deliverable)
- External support delivered by Agile coaches (change deliverable)

We use three types of deliverables. **Technical deliverables** are classic deliverables such as a role description, a process, a product or an IT system. These deliverables are used by people to work in a new way. **Training deliverables** that support acquiring the competences each role needs and **change deliverables** that ensure that the necessary change in behaviour actually happens.

Later, we will use an example of how to formulate the change deliverable “implementing daily stand-ups in an Agile team” by writing it as a user story. This user story can point to one or more of the elements in the benefit map, in order for us to know exactly what kind of behaviour we aim to gain by working with the specific change.

Using benefits realisation going forward in the change

Before making the final decision about Agile throughout the organisation, we develop the benefit map from a sketch on the wall to our common image of what we need to do, how we must work and what

benefits we want and why. On top of that, we build measures for benefits as well as behaviours, so that we know the kind of behaviour that is critical for us to realise the benefits.

By building a benefit map, we create management ownership and a demand for implementation, and we acquire important input to the story about the change. The benefit map also provides important initial input about what we can be flexible on and what will be a “non-negotiable” when it comes to behaviour for the different employee groups.

Estimating the benefits is difficult, but it is an even more difficult task to estimate the change effort. But we need these estimates to find out whether we believe that Agile will work for us in more than just one small corner of the company.

To estimate the change, you need to understand how to implement Agile in a brain-friendly manner and how to use an Agile approach while doing it.

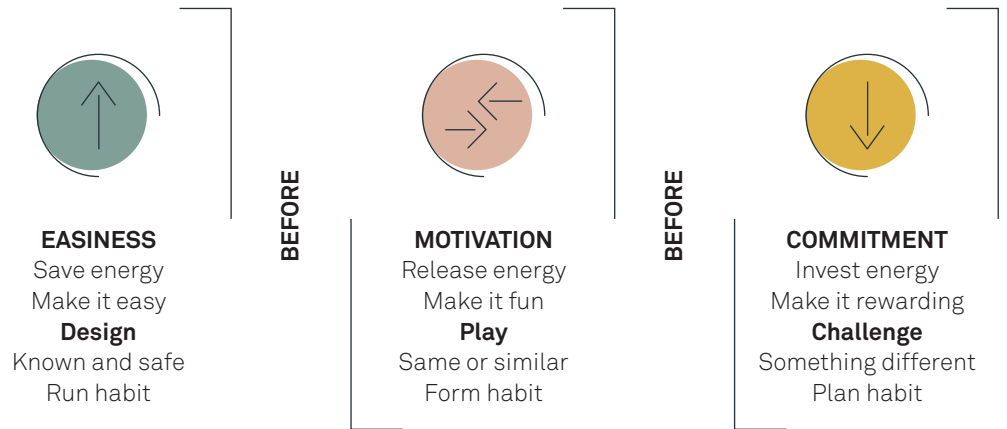
Making the Agile transformation brain-friendly

Failing to change behaviour is the most important reason why most transformation projects do not realise the benefits they were intended to. In short, realising our benefits in an Agile transformation depends on our ability to change our behaviour.

By using the Brain-Friendly Approach, the chance of creating behavioural change is much bigger. We will not delve into the reasons why the Brain-Friendly Approach works, but merely suggest further reading on the topic from our colleague Laust Lauridsen. Here, we will describe the principles of the Brain-Friendly Approach that we find relevant to Agile transformation, and very briefly describe how we apply this in Agile transformation.

After that, we will unfold the specific elements in the Agile approach and mention the Brain-Friendly Approach when relevant.

Figure 3
Source: Laust Lauridsen / The Brain-Friendly Approach



The Brain-Friendly Approach: “Make it easy, fun and rewarding”

Using an Agile approach to an Agile transformation frequently reminds us to make it “easy, fun and rewarding”. We use it as a guideline on how to think and design the change. And we use it in our daily work. There is no big, easy answer to it, but a lot of small ways to constantly have that approach implemented in the Agile change. We will unfold this more in the “Agile approach” part of the article.

The Brain-Friendly Approach: “The brain reacts positively to clear goals” and “use time to define the wanted behaviour”

In the benefit map, we describe the wanted behaviour on an overall level. But we are much more specific when we use the Agile methods for change. By writing user stories with acceptance criteria and breaking the actual behaviour down into specific tasks in the design of the actual change, we aim at having very clear goals that are easy to train.

The Brain-Friendly Approach: “Make change on demand” and “make change pull instead of push”

Becoming Agile is a large change for all individuals involved. But we can create a pull towards change instead of a push, for instance by using a Kanban board with all behavioural changes of the people involved in the change, and by letting

them choose for themselves what they want to focus on changing next in the Agile transformation.

Agile transparency in the Agile transformation

In Agile transformations, we often see that organisations lose their bearings in the change and from time to time feel that the Agile change and behaviour stands still or even goes backwards.

When implementing Agile methods and mindsets, we suggest using an Agile approach. By doing this, we ensure, among many other things, transparency of the specific elements in the change, and we believe it is easier to apply the Brain-Friendly Approach, as described earlier.

To unfold this in more detail, we will go through a concrete example of how to use an Agile approach to an Agile implementation by writing user stories, breaking user stories down into tasks and having stand-up meetings with key stakeholders at fixed intervals.

The example will be about implementing daily stand-ups in a permanent team. The reason for using daily stand-ups as an example is that it is a central ceremony that is easy to understand, but difficult to implement to an extent where the team in effect is self-organised around motivated individuals.

Agile changes formulated as user stories

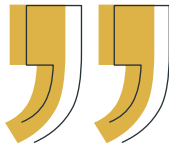
If we use the structure in a user story, “As <role / person>, I want <what?> so that <why?>” it ensures all the advantages that we normally gain by using user stories to formulate what we want. It is very precise about who needs to do the change, what we want to do and why we want to do it.

Let’s see what a user story with our daily stand-up could look like:

- **Headline:** “Daily stand-up”.
- **Description:** “As a Scrum Master, I want to facilitate daily stand-ups, so that team members co-ordinate with each other and issues are identified.”

As with all user stories, formulating the acceptance criteria is essential. They enable us to see when a task is done and make it easier to break the user story down into specific tasks.

Examples of acceptance criteria to our specific “daily stand-up user story” could be:



- “The team talks to each other and follows the structure: What have I done since our last stand-up, what am I going to do until the next stand-up and do I have any issues?”
- “Issues are identified at the stand-up and solved after the meeting.”
- “The team takes responsibility for user stories and tasks on the board.”

We could write it with the team as the role/person, but for now, we will focus on the behaviour of the Scrum Master, and what he/she can do. When we coach and train an Agile transformation while using the Brain-Friendly Approach, it is essential to stress that it is the person(s) who is undergoing the change that writes the user story. It will often be with help from an Agile coach who has actual experience with the concrete Agile behaviour.

This will be one of many user stories in an Agile transformation. To have a good

stand-up, we – at the very least – need to have a good refinement process with “Definition of Ready” (DoR) criteria and a good planning session as well as a set of rules to live by in the team. These will have their own user stories (change deliverables) to be delivered in the change effort.

Agile changes as specific tasks

When we write change behaviour as a user story, it is the first step in the Brain-Friendly Approach because we make it specific and easy. But we want to make it even easier and even more fun and rewarding by breaking our user story down into specific tasks. When we do that, we decompose the behaviour and are encouraged to be very operational with the specific behaviour we want to create.

When you write down the tasks of the new behaviour, we encourage you to ask yourself: Is this easy? How can we make this fun? How can we make it rewarding?

Specific tasks for our daily stand-up user story could be:

- Talk with the team about when to have the daily stand-up (so they can commit to the specific time) and invite the team to the daily stand-up
- Write the three questions down and hang them up on the Scrum board (so we remember them)
- Ask the team to inform each other at the stand-up (and not to report to the project manager or the product owner)
- Ask team members to move their own tasks (so they also undertake responsibility for reporting the task)
- As a project manager, move away from the inner circle (so team members don’t report to you)
- Think about your own energy when facilitating the stand-up – have fun
- Place sweets on tasks as rewards when moving to “Done”

And so on. A good stand-up will require more tasks. Experienced Scrum Masters know how many details need to be in place to have a daily stand-up where teams are encouraged to work together and discouraged from reporting to a proxy project manager instead of working together on common goals.

Transparency in the Agile change

By now, we have a user story that is broken down into specific tasks. To make it visible to our new Scrum Master, the team and everyone else involved in the change, we will use a Kanban board to create transparency in the change. We will have all user stories and their specific tasks on the board. In the beginning, everything will be in “To Do”. In Agile, this is business as usual.

Because we cannot work on everything at one time, and because we need to train in order to master a specific behaviour, we will start with two to three tasks from our “To Do”. We will only work on tasks in “Doing” and move tasks to “Done” when we can see that the actual change in behaviour has been implemented. Furthermore, by doing this, we can manage WIP (Work in Progress) limits on the Kanban board, ensuring focus on a few things and hereby, maximising the possibilities of experiencing flow. It is also important to let the person(s) involved in the specific change prioritise which tasks to work on and decide when to start a new task.

When you have an overview of the Agile transformation, it also ensures good insights into the status of the overall change and not only the specific behavioural change related to one user story.

Stand up to the Agile change

When we make our progress visible to everyone involved in the change, we exchange information about the elements in our new behaviour that we are working on right now. By meeting with key stakeholders at fixed intervals, we ensure

that everyone is on the same page at the same time. There are several stakeholder groups in an Agile transformation that will benefit from knowing: What we are working on right now, what kind of specific challenges we have and how we are handling them.

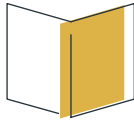
Zooming back into our “daily stand-up user story”, we suggest that the Agile coach has a stand-up/status meeting with the new Scrum Master and the team once every two weeks to talk about the kind of new behaviour currently being trained. This could be in retrospect to each sprint.

When all tasks are moved from “To Do” to “Done”, we can look at each of our user stories and see if we meet the acceptance criteria of the user story. If we do that, we can move the user story to “To Do” and give high fives or other rewards to the ones involved in the specific change.

Next step Agile

An Agile change is a large behavioural change. In this article, we have only covered a small corner of it. Are you undergoing or thinking about working with Agile methods? Then, an operational next step could be to ask yourself the following questions:

- Do we have a purpose working with agile?
- What kind of benefits are we trying to achieve?
- What kind of behaviour do we want?
- Do we have transparency in our activities?
- Is the change in behaviour specific?
- Do we have an agile approach to the change?
- Are we working on making it easy, fun and rewarding?



Suggested further reading

The Brain-Friendly Approach

by Laust Lauridsen

<https://implementconsultinggroup.com/the-brain-friendly-approach/>

Benefits realisation

by Rasmus Rytter

<https://www.linkedin.com/pulse/key-tool-benefits-realisation-rasmus-rytter/>

FAST FACTS ABOUT IMPLEMENT

Founded: 1996

Number of employees: 800

Headquarters: Copenhagen

Offices: Stockholm, Malmo, Oslo, Zurich and Munich

implementconsultinggroup.com

About the authors

Rasmus Rytter, Management Consultant, Implement Consulting Group

Rasmus has in-depth subject matter expertise in benefits realisation projects. He works across business and IT and complements hands-on and advisory jobs with training project managers in “Advanced project management”, “Change management” and “Benefits realisation”.

rry@implement.dk
+45 30 85 80 63

Christian Mark Christensen, Management Consultant, Implement Consulting Group

Christian has significant experience in planning and executing Agile transformation. He is an expert in Agile work methods (SAFe, Scrum and Kanban), organisational Agile design and servant leadership.

cmch@implement.dk
+45 23 71 27 68