

The Capgemini Academy agile transformation approach

Academy



Becoming agile is now indispensable for most companies. The question is no longer “Do we need to do this?”, but “How are we going to do this?”

Agility means being able to respond faster to evolving markets and evolving technology. This transition is accompanied by and fused into the digitalization of organizational processes. There is a trend towards incorporating more automation, robotics and artificial intelligence in processes to enhance speed as well as quality, and to reduce costs. Organizations’ primary processes will be performed with increasing pace by automated systems and robots, together with the supporting processes, such as HR, legal and finance. More and more process steps will be performed by IT and robots instead of humans. Being able to enhance, improve, adjust and replace processes quickly is imperative these days. The improvements to our primary processes need to happen faster and better. Put simply, if you do not innovate fast enough, you will become obsolete.

The adoption of an agile transformation is a complex program that encompasses three main areas:

1. Organizational change
2. Cultural change
3. Technical change

In this white paper we will outline the steps needed in the various areas of an organization to become agile. Although many of these steps will happen concurrently, others will happen sequentially, as the organization becomes more mature in agile ways of working. Organizations in which IT is an essential part of their primary processes will find this white paper most useful.

| Areas | High-level activities | | | Timeline |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|
| 1. Transformation team | Form the transformation team. Train change agents. | Create the transformation vision, strategy and roadmap | Create a lean-agile center of excellence. | |
| 2. Organization | | | | |
| Management | Train management in agile mindset and leadership. Assign product owners with mandate. | Repeat the message frequently. | | |
| Team | Set up Scrum teams. Assign agile coaches. | Scale up the number of agile teams. Set up the first DevOps teams. | Coordinate the work of various teams in value streams. | Increase the number of value streams with agile teams. |
| Individual | Train people in agile methods. | People work in self-organizing teams. | People become T-shaped. | |
| 3. Cultural | An ongoing process of communication of the agile mindset and values. This includes creating suitable office spaces, decorating offices, storytelling, and sharing successes and failure. HR has an important role in guiding the hiring process and performance measurement of individuals and teams. | | | |
| 4. Technical | Start working on automated testing. Work with architects and plan for agile architecture. | Define metrics and increase measuring of performance on all levels. | Work on automated testing, automated builds, continuous integration and delivery. Improve system and software architecture. | Improve automation of the delivery pipeline. Automate end-to-end testing. Automate provisioning. |

Sense of urgency

In order to make change happen and to make it successful, we need a sense of urgency. People need to feel that change is important for the future of the organization. Even if we are doing quite well as an organization, we need a visionary statement that shows people that change will make the world a better place. Change can ignite from a situation of accumulated problems, incidents, complaints and despair and the sense of urgency will likely have been growing stronger over the past years. However, if you find yourself in this situation, it might be too late.

Change can also grow as an idea based on a strong vision, where we see possibilities of getting ahead of our competitors by making changes now, even though we are doing well. Change can start in the lower ranks of the organization and it can start at the top, when a visionary leader foresees the required steps and creates inspiration.

The reasons for becoming an agile organization are:

- Since the adoption of smartphones, tablets and apps, the development cycles of new features have shortened tremendously, and customers expect rapid change and improvement.
- There is a rapid increase in new technologies that have high potential for disruption, such as blockchain, IoT (Internet of Things), VR (virtual reality), AR (augmented reality), artificial intelligence, cloud solutions and quantum computing.
- There is a proliferation of start-ups using technology to create new business models. They disrupt current business models and pose serious competition. Large, traditional organizations such as banks, insurance companies, telecommunications providers and government institutions must respond to this tidal wave of innovation. Fintechs in particular pose an interesting challenge for banks. There is a vast increase in companies that offer financial services. And the sleeping giants, such as Google, Amazon and Alibaba might also one day start their own banking services.

The goal of change

Becoming an agile organization should have two main objectives:

1. Reduce your release cycle time; how fast can you get new ideas into production?
2. Increase customer value and business value; always strive to deliver the highest customer value that also enables higher business value.

There can be a third objective, which, however, should not be your main focal point: reducing operational cost. If agile methods are used correctly, the cost of operations and projects can certainly be reduced. In the past many IT projects were developed in waterfall stages: first requirement definition, followed by design, building, testing and finally implementation. Many of these projects were seriously challenged. Agile methods carried out correctly will deliver more customer value faster and will deliver better value for money.



The agile transformation team

Becoming an agile organization is a huge transformation program that can take many years and will require large investments and sacrifices. A specialized team is required to lead such a transformation, with indispensable commitment and involvement from top management, business and IT departments. Cultural change coaches are needed alongside specialists in Scrum, kanban, lean, scaled agile methods and DevOps. Architects are needed to adjust the landscape of systems, in order to enable autonomous teams. The agile transformation team also works in an agile way by creating a vision, a backlog and a transformation roadmap.

Organizational change

Start forming multidisciplinary, full-time, dedicated agile teams and have teams practice with Scrum and kanban. Gain experience in agile ways of working. In certain areas it will be fairly easy to form successful Scrum teams, such as those working on front-end applications, without many dependencies with other departments. In other departments it will be harder to form autonomous teams, because of the dependencies with other teams. The goal of the agile teams is to deliver customer and business value by building a shippable product increment at the end of each sprint. This is a journey that they will make with many improvements along the way.

During the sprints, there should be increased cooperation with the system operations teams and at a certain point it should be possible to form DevOps teams, where all development as well as operational tasks are performed. These teams will have end-to-end responsibility for a service or product. They build, test, deploy new features into production and maintain them as well.

Create various flavors of teams: some will be responsible for services and features that will be used directly by end-users (called feature teams), while others build the underlying infrastructure, tools and routines that can be used by the feature teams. The latter are often called platform teams.

Teams usually have several dependencies with other teams, either because they work on the same product or because they use the same resources. This makes changes to systems much harder, as it requires a lot of up-front coordination. We try to make teams as autonomous as possible. To prevent endless coordination efforts ending up once more in waterfall staged projects, we need to make sure that agile teams can work autonomously but still coordinate sufficiently with each other to manage dependencies and perform frequent integration of their deliverables.

In the past we initiated new projects and change initiatives by defining business cases and project plans and projects could start after formal approval by a project board. Now, instead, we visualize many small changes in a product or service. You could consider these as mini-projects. The primary process for delivery of a product or service is called a value stream. Let us take a "mortgage" value stream as an example: all the steps that are required for a bank to deliver its mortgage product to a customer. In the agile way of working we define the products or services in which we need a certain amount of innovation and we view our total available innovation budget. We then define how much budget is available for the "mortgage" value stream and we set up dedicated agile development teams. Teams can be organized around products or services, but also around systems that handle transactions for several value streams. In this way the budget is "fixed" and the teams must strive to deliver as much value as possible in the shortest amounts of time. We then have budgeted value streams instead of projects. Also consider the role of management in decision making during development. Define moments when official approvals are required. Redefine the way auditors and change advisory boards work, so as to avoid inhibiting the agile teams while still retaining a certain level of control. Define how to make the systems auditable.

Spread the news

Once you have gained experience and you have celebrated your first successes it is time to spread the rumor to other teams and have them form agile teams as well. For some teams Scrum will be a great method, whereas for others kanban will suit better. For some changes traditional project management will be perfectly fine. If you feel it is important to receive frequent feedback from users and other stakeholders during demos, Scrum is probably a good method for you, but if the nature of work is more characterized by short, rapid fire, ticketed tasks, then Kanban might be the way to go.

Many people will not be convinced that all this reorganization is necessary or that this new way of working is better. Many people need “proof” that all the fuss is worth it and need to understand the reasons why we are doing this.

Training

As the news spreads, people need to be trained in agile methods and role fulfillment. There will be new roles such as Scrum master, release train engineer and product owner and people need to have a thorough understanding of what these roles encompass. They benefit from workplace coaching. Agile team members need to be able to fulfil various roles, or at least have a deep understanding of other roles. This will enable them to assist others in their tasks and to solve problems together, which will lead to more team autonomy.

Top management should lead the way

A crucial role is laid out for top management. Commitment from higher management is crucial for the success of the agile transition. Strategic, difficult decisions need to be taken, and top management needs to initiate the cultural transition that is required. Cultural change is empowered from the top and not from the bottom of the organization. It is important that C-level roles thoroughly understand the consequences of the agile way of working. This might be hard: management will have to delegate more responsibilities to product owners and agile teams. They will have to stop demanding projects with fixed time-lines, fixed budgets and fixed scopes. They will have to start thinking in terms of product-based budgets instead of project-based budgets. Top management should lead the way by setting the right example. One concern for management is losing control with agile ways of working. However, because of the cyclical nature of development, with reviews and demos at the end of each sprint, management is able to give feedback, control and adjust what the development teams will work on in the next sprints. Instead of losing control, management can check progress and adjust development much more frequently.

The role of management in an agile organization should be:

1. Vision development and setting strategic goals
2. High-level, strategic decisions
3. Enabling autonomous teams
4. Enabling shared consciousness through¹:
 - Extreme participatory transparency (e.g. open physical spaces; holistic awareness)
 - Strong internal connectivity across teams (create connections between teams by embedding and liaisons)

¹ From: Team of Teams: New Rules of Engagement for a Complex World – May 12, 2015 by General Stanley McChrystal, Tatum Collins, David Silverman, Chris Fussell

Cultural change

For an organization to become agile it must have small, high-quality, autonomous teams who can take decisions swiftly and accurately. Organizations become flatter and management should facilitate and enable these autonomous teams. Teams do not work in isolation, but communicate and plan together, so they can keep a holistic view of the organization. An open culture is required where teams can experiment, self-organize and learn quickly from each other. Teams should take joint responsibilities and there should be a culture of sharing knowledge.

Failure

There is a lot to do in the agile world around failure and making mistakes. An important aspect of an agile culture is the way we deal with mistakes. To be clear, making mistakes is not acceptable if the consequences are “lethal” to the company or if we do not learn anything from them. It is easy to say to employees that they must experiment and become more autonomous, but that is going to be very hard if the effect of a mistake is detrimental to the company. In a fast-paced, complex world, it is important that teams are allowed to experiment and test their ideas. It should be possible to make mistakes, even if sometimes the consequences are bigger. Many small mistakes probably create a bigger learning experience than one huge mistake. Importantly, mistakes should never be a “one-person” fault. If we work in teams and we own up to our responsibilities, we also own up together to a mistake that one team member made. We find ways to make our systems and organization more resilient, so that one mistake will not topple the whole company.

Technical change

Teams who work on IT systems also will need to improve their delivery pipeline: the process with which they create and deliver new software. Teams will need to start working on practices such as automated testing, peer reviews and pair programming. This enhances their speed of delivery as well as quality. Test as early as possible and invest time and money in automated testing. Create central repositories, where developers check in their code and have automated tests carried out. Develop tools for automated builds, automated testing, automated provisioning, and automated deployment into production. Ideally, software engineers should have the possibility of creating a production-like environment on their own laptop. Automate as much as possible and create continuous delivery pipelines. In these pipelines developers do not need to wait for the end of a sprint before deploying into production, but can deploy much more frequently. Set up high-quality source control and configuration management. Software and systems will change versions all the time, and this should be managed automatically. In this way, it eventually becomes possible to create any environment at the click of a button.

One important task is to create metrics. By measuring performance and behavior we can create feedback loops, not just to business people but particularly also to the agile teams. The only way to really become agile is to experiment often, receive feedback from real live systems and customers, test your hypothesis, learn and improve. This implies that you need to create metrics on all levels in your stack: customer behavior, application, middleware, databases, servers, platforms, networks, security and other items that you consider important.

Architecture enables autonomous teams

One of the factors that can seriously hold back teams from becoming autonomous is the architecture of our IT systems. In many companies these systems have grown steadily not just in size but also in the variety of systems with many interfaces with other systems. If teams still need to coordinate with several other teams before they are allowed or able to make a change to one system, you need to start working on refactoring your systems architecture. As your organization becomes more agile, your system landscape needs to become agile as well. In fact, architecture is one of the best predictors of agile team performance². The more agile your architecture, the more autonomously the teams can work and the faster they can deploy new features into production. Architects need to start designing a strategy to move step by step (or by means of a big bang) to a new agile architecture. Part of this architecture will probably be based on SOA (Service Oriented Architecture) or microservices. The sooner you start working on this the better.

In general, you will automate an increasing proportion of your primary processes, and you will work continuously on making these processes more effective and efficient. You will use multidisciplinary teams to create new solutions and solve complex problems. Some specialist “swat” teams will swarm from problem to problem and deal with them quickly. Others will be formed as needed for short periods of time. You must be able to act quickly.

Finally, these are the knowledge areas and competencies that are required for agile transformation:

- Cultural change
- Lean theory
- Kanban
- Scrum
- SAFe (or a similar scaling method, for large, complex systems)
- DevOps
- Coaching and facilitation skills
- Agile leadership



² 2015 State of DevOps Report, by Puppet Labs.

For more information contact:

Author

Rik Pennartz

Email: rik.pennartz@capgemini.com

Phone +31 6 1067 1189

Reviewers

Pieter de Beijer

Addo de Visser

About Capgemini Academy

Capgemini Academy's professionals offer what people in IT need. Our professionals have a keen eye for motivation, talent and are aware of specific contexts and circumstances. They move people to move. Programmes and courses that originate from daily experience of our both didactical and substantively strong trainers, light a fire within the individual IT professionals. Real life stories of our professionals' experience that tell how to solve problems and work with the people around it, do the rest.

An organization, like ours, helps people and their organizations day by day to get the best out of themselves and each other. We prepare them to defy tomorrow's challenges. We stimulate learning and curiosity. In order for individual IT professionals and their employers, to build better, longer and more intensive relationships. For mutual benefit.

Capgemini Academy. We transform IT professionals.

academy.capgemini.nl

The information contained in this document is proprietary. ©2018 Capgemini.