

IT AGILITY EXPLAINED: ACHIEVING AGILITY ACROSS THE ENTERPRISE



The demands for businesses and IT to be quicker in responding to the ever-evolving customer and operating environment do not seem to be slowing anytime soon. Among the leading forces of change in the tech space [McKinsey](#) lists:

- Digitization
- Globalization
- Automation
- Analytics

And then COVID-19 joined the game and massively disrupted operations while accelerating the digitization journey for almost all organizations. This of course forced laggards who had never considered remote working, cloud, apps and social media to immediately get on the bandwagon or risk oblivion from the unexpected pandemic risk materializing.

In the ever-evolving world of information technology, being able to speedily, yet effectively, respond to market changes is a difficult but critical task.

As organizational capabilities change, so must IT capabilities—and sometimes it becomes necessary to reconfigure or completely replace organizational structures, processes, or systems in response to evolving marketplace realities. At the same time, though, there is still a desire for control and stability, hence any lingering hesitation to embrace all change in IT environments.

An increasingly common suggestion for how businesses can achieve all these things effectively is through IT agility.

What is IT agility?

In general, agility is a common business term that refers to how fast an organization responds to opportunities. It is typically recognized as the time in between an organization becoming aware of a potential business opportunity and acting on it.

The [ITIL® 4 Foundation publication](#) defines organizational agility as the ability of an organization to move and adapt quickly, flexibly, and decisively to support internal changes. This could include:

- Strategy, practices, or technology requiring different skills or organizational structure
- Changes to relationships with partners and suppliers

IT agility, then, is a measurement of how efficiently the IT infrastructure of an organization can respond to external stimuli.

This can mean how effectively it embraces the pressure to change or how successfully it creates a new opportunity. Instead of being thought of as another task to complete, IT agility should be viewed as more of an overall mindset, eventually becoming part of the company culture.

While there are many approaches to IT agility, [the agile manifesto](#) has been associated as the go to reference in the world of software development, by embracing frequent delivery and welcoming changing requirements. As opposed to ["waterfall" methods](#), an iterative incremental delivery approach is applied through bi-weekly or monthly sprints. At the end of each sprint, the work and project priorities are evaluated, which allows for client feedback to be incorporated, as well as including improvements and changes.

(See how [agile & service management](#) work together.)

Agile	Waterfall
Iterative development in short sprints	Sequential development process in pre-defined phases
Flexible and adaptive methodology	The process is documented and follows the fixed structure and requirements agreed in the beginning of the process
Feedback-based approach: Sprints lead to short build updates that are evaluated on and guide the future direction of the development process.	Limited and delayed feedback: The software quality and requirements fulfilment isn't evaluated until the final phase of the development processes when testers and customer feedback is requested.
A provision for adaptability: Project development requirements and scope is expected to change over the course of the iterative development process.	The requirements and scope are definitive once agreed upon.
The SDLC phases overlap and begin early in the SDLC: planning, requirements, designing, developing, testing, and maintenance.	The SDLC phases are followed in order, with no overlap. Members of one functional group are not involved in another phase that doesn't belong to their job responsibilities.
Follows a mindset of collaboration and communication. The requirements, challenges, progress, and changes are discussed between all stakeholders on a continuous basis.	Follows a project-focused mindset with the aim of fully completing the SDLC process.
Responsibilities and hierarchical structure can be interchangeable between team members.	Fixed individual responsibilities, particularly in management positions.
All team members focused on end-to-end completion (achieved sequentially) for the projects.	Team members focused on their responsibilities only during their respective SDLC phases.
Suitable for short projects in high-risk situations.	Suitable for straightforward projects in predictable circumstances.
Limited dependencies as the focus is less on implementation specifics, and more toward the mindset.	Strict dependencies in technologies, processes, projects and people.

Principles of IT agility

There are a variety of common [principles](#) that can be gleaned from the agile manifesto in IT agility including:

1. Satisfy the customer through early and continuous delivery
2. Deliver updates frequently through bi-weekly or monthly sprints
3. Cultivate an environment of changing requirements
4. Pay special attention to technical excellence and good design
5. Promote strong communication between business people and developers
6. Keep it simple
7. Encourage continuous reflection on progress as well as what improvements can be made

But one must be careful not to implement IT agility in a manner that is at the expense of business outcomes and value. A fragmented approach can create other bottlenecks in the flow of work resulting in frustration and misalignment, cost overruns, [high technical debt](#), and ultimately unhappy customers.

In addition, speed sometimes can come at the expense of completion and quality. A holistic approach is more prudent, tapping into organizational strategy and customer experience at the heart of it.

The [Harvard Business Review](#) gives six principles for building a company's strategic agility that could also be considered as the premise for IT agility:

- Prioritize speed over perfection
- Prioritize flexibility over planning
- Prioritize diversification and "efficient slack" over optimization
- Prioritize empowerment over hierarchy
- Prioritize learning over blaming
- Prioritize resource modularity and mobility over resource lock-in

How can you achieve IT agility?

It is important to remember that IT agility is not a quick project that can be executed over a long weekend; IT agility requires an entire shift in the company's ethos and thinking.

(Learn about [Lewin's three steps to change](#).)

Once everyone is onboard with this change, an evolving plan should be put into place to map out short-term and long-term strategic goals. If you have a solid outlook on where you would like your systems to be going, then it makes it that much easier to select the appropriate opportunities that will get you there once they come around.

To begin creating this plan, the business must first reflect on some of the key factors that are driving the application of agility to begin with:

- Are systems tightly coupled and opposed to change?
- Are deployment schedules constrained due to testing complexities and integration dependencies?

By finding out what is encouraging the change, it will help goals be specific and relevant.

Sure, becoming more agile might seem like an extensive plan and complete overhaul of the system is needed. But keeping it all as simple as possible is an important mantra to remember.

For the most part, re-purposing components and systems that are well-designed will be a quick and consistent way to cover the various problem spaces. However, where there exists rigidity in [IT infrastructure](#), it will always be a stumbling block—[the digital age](#) cannot wait for traditional approaches to acquiring and provisioning of underlying components to support software.

Migrating to the [cloud](#), [infrastructure automation](#), [CI/CD technologies](#), and [SRE practices](#) can help deliver the flexibility and self-service capability needed by developers to get the infrastructure they require on demand.

People are at the heart of any transformation. IT agility is no exception.

Hierarchical structures with their 'command and control' can end up limiting speed when it comes to decision making and allocation of resources. The same can be said of [traditional change advisory boards \(CABs\)](#) that stand in the way of quick decision making.

Transforming to matrix structures that are adept at quick allocation/reallocation of resources to priority needs is preferable. In addition, permanent, simple multi-competent teams that are assigned to work exclusively on a product can provide the autonomy needed to both:

- Make faster decisions
- Deliver features and solutions quicker

In this holistic approach, other elements throughout the value creation and preservation journey must be considered. IT agile requires agility in budgeting, contracting, procurement, and [any other practice that is involved in IT activities](#).

This requires not just agile processes—agile mindsets are more importantly.

Agility is a mindset, not a switch

IT agility is about far more than just adopting new strategic plans and development practices. It requires an entire rethinking of the IT organization to successfully meet the intended goals and move one step closer to complete enterprise [digital transformation](#).

Starting efforts towards an agile IT is not easy, but once the process gets underway, many organizations see improvements extremely fast. Changes early on can free up resources, which allows IT to better support the digital transformations and improve further developments.

There is a direct correlation between having an agile IT and increasing the Time to Value (the time between an initial request and its delivery) for a business. IT agility is no longer the wave of the future but an immediate imperative: are you ready?

Related reading

- [BMC Business of IT Blog](#)
- [Best Practices for Organizational Change Management](#)
- [What Is "IT-Business Alignment"?](#)

- [The Scaled Agile Framework \(SAFe\): What To Know and How To Start](#)
- [Digital Transformation: 5 Tips for Success](#)
- [Application and Software Modernization: Concepts, Pros/Cons & Best Practices](#)