

# The Differences Between Scrum as a Service and Traditional Contracting



## The Story of the Ship and the Iceberg

Once upon a time, a ship was cruising majestically in the Atlantic Ocean.

Halfway through the journey, the ship's crew spotted a white object floating on the horizon. The object seemed almost ridiculously small at the beginning but as they got closer, the enormity of this white mass started to become apparent. The ship was heading straight towards an iceberg, and an enormous one at that!

Panic-stricken, the crew members rushed to the captain's quarters and broke the news. But to everyone's surprise, the captain commanded them to stay on course. "We can't change course now fellas... This is the trajectory that we agreed upon earlier and we're going to stick to it no matter what!"

A few minutes later...

3...2...1... BOOM!

The End.

So... What did you think about the story?

Were you surprised? Agitated maybe? The captain's response seemed absurd, didn't it?

But guess what? This is exactly what happens during the software development process in traditional client/contractor partnerships. Teams involved in these projects often set sail hoping to achieve success and end up crashing into massive icebergs that they couldn't foresee when they began their voyage.

You see, the development of innovative software solutions requires an Agile mindset. Teams need be able to change course and flex on scope when necessary, but they must also be able to do it efficiently i.e. in a way that minimizes the cost of change and maximizes innovation.

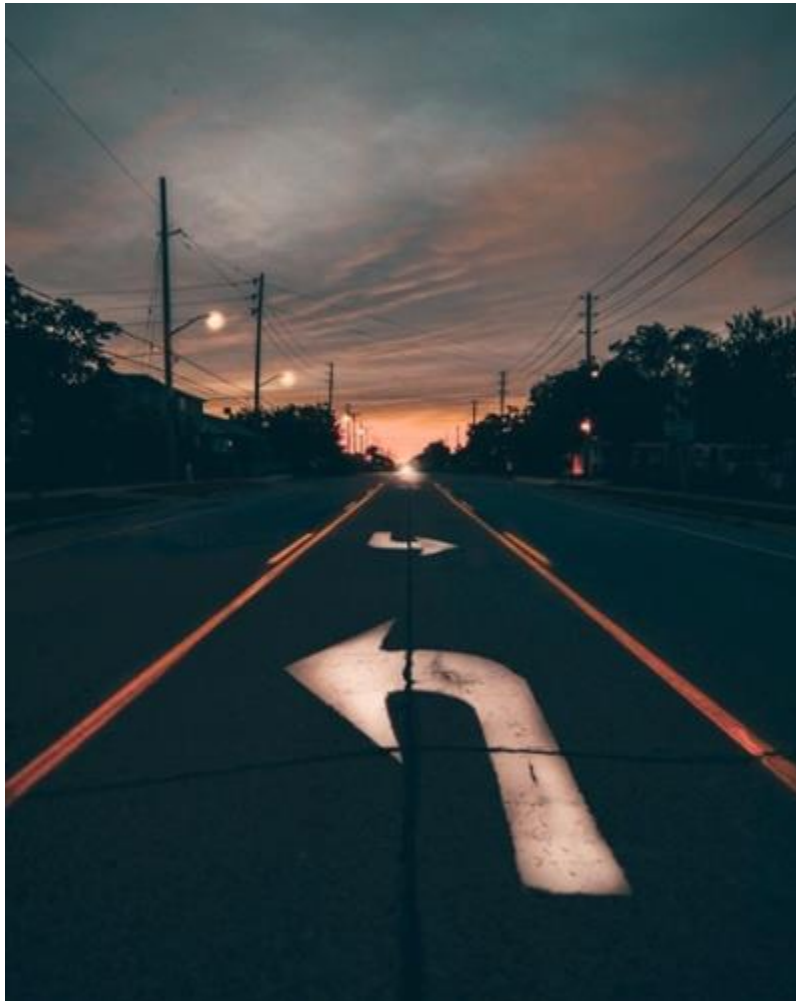
And that's exactly what [Scrum as a Service](#) achieves.

## The Differences Between Scrum as a Service and Traditional Contracting

1. Flexibility vs. Rigidity
  2. Product Success vs. Contract Requirements
  3. Aligned Incentives vs. Lack of Partnership
  4. Learning from Adversity vs. Avoiding Failure
  5. Rapid Innovation vs. Reduced Creativity
- [Looking for a world-class Scrum team that's ready to help you turn your vision into high-quality working software?](#)

Now, let's look at each of these in detail...

# Difference #1- Flexibility vs. Rigidity



Traditional contracting fixes the trajectory of the development process and leaves little room for change. And this is a big mistake.

As mentioned previously, the development of innovative software products is similar to a cruise in an ocean filled with scores of icebergs. Some huge and some tiny.

The point is, you can't see all of them right from the start. And in order to arrive successfully at your destination, you'll need to be able to gauge the situation and change course effectively.

[Scrum as a Service](#) is an approach that allows you to adjust your approach and flex on scope as you go. Scrum teams understand that it's impossible to gather all the requirements at the beginning of a project. This means that not only do team members expect change, they embrace it.

This adds a huge degree of flexibility to the innovation process and allows teams to cut costs and adapt to almost any situation that presents itself.

[Looking for a world-class Scrum team that's ready to help you turn your vision into high-quality working software? >>](#)

[Back to Top](#)

## Difference #2- Product Success vs. Contract Requirements



Through traditional contracting, project managers often try to bind contract requirements to the product's success. But this rarely turns out to be a successful approach... especially when it comes to software innovation.

Countless projects in the past have failed miserably once deployed to the general public even though they were on time, on budget, and on scope. These failures happened primarily due to the misconception that the original scope of the project accurately captured what the public wanted.

The teams involved in these projects didn't bother to check how the public's needs evolved over time and ended up developing the wrong product.

This is a fairly common issue. An astounding number of teams realize – often when it's too late to turn back – that there's a dire issue with their initial planning and that the product's scope needs to change. And there really isn't much they can do about it. They are bound by the contract requirements and feel obliged to abide by them.

[Scrum as a Service](#), on the other hand, is an approach that puts the product's success before everything else. Scrum teams understand that at the end of the day, it's all about the software and what it does for the client. This allows them to channel their efforts and focus on the rapid development of a successful MVP (Minimum Viable Product).

[Back to Top](#)

## Difference #3- Aligned Incentives vs. Lack of Partnership



Rigid contracts often lure contractors into thinking that they are merely here to accomplish a piece of work and not to share responsibility for the entire project's success.

This lack of partnership poisons the collaboration between the various domains (e.g. development, infrastructure, marketing, etc.) involved in the project. The result? Simple. Failure once all these domain solutions have to integrate.

This is something that does not happen in the [Scrum as a Service](#) approach. Once hired, Scrum teams become an extension of the client's company. Team members know that they succeed only when the client succeeds.

## Difference #4- Learning from Adversity vs. Avoiding Failure



Traditional Contracting regards failure as a sinister happening. Something that needs to be avoided at all costs. This attitude adds additional pressure and limits the team's ability to explore new possibilities and innovate effectively.

Software, by its very nature, is an uncertain science. And this uncertainty only increases with the complex and novel nature of innovative projects.

[Scrum as a Service](#) embraces this uncertainty. Failure is seen as a learning opportunity. An inevitable and often essential stepping stone towards success.

Scrum teams are also firm believers in the Fail Fast Early method. Team members have the freedom to fail fast, learn from these failures and adapt quickly to the ever-changing circumstances.

# Difference #5- Rapid Innovation vs. Reduced Creativity



Through traditional contracting, project managers and clients often complain about the lack of creativity and innovation. But they rarely realize that confinement is the enemy of creativity.

Rigid contractual costs and schedules end up becoming thick filters that inhibit the flow of creative thinking within the team.

[Scrum as a Service](#), however, creates the ideal environment for innovation. Scrum teams have the freedom to explore new ideas and take calculated risks to accelerate the development process.

This freedom to explore, coupled with the desire to succeed makes Scrum teams unbeatable at developing highly-innovative software products.

Looking for a world-class Scrum team that's ready to help you turn your vision into high-quality working software?