BASE2
SERVICES

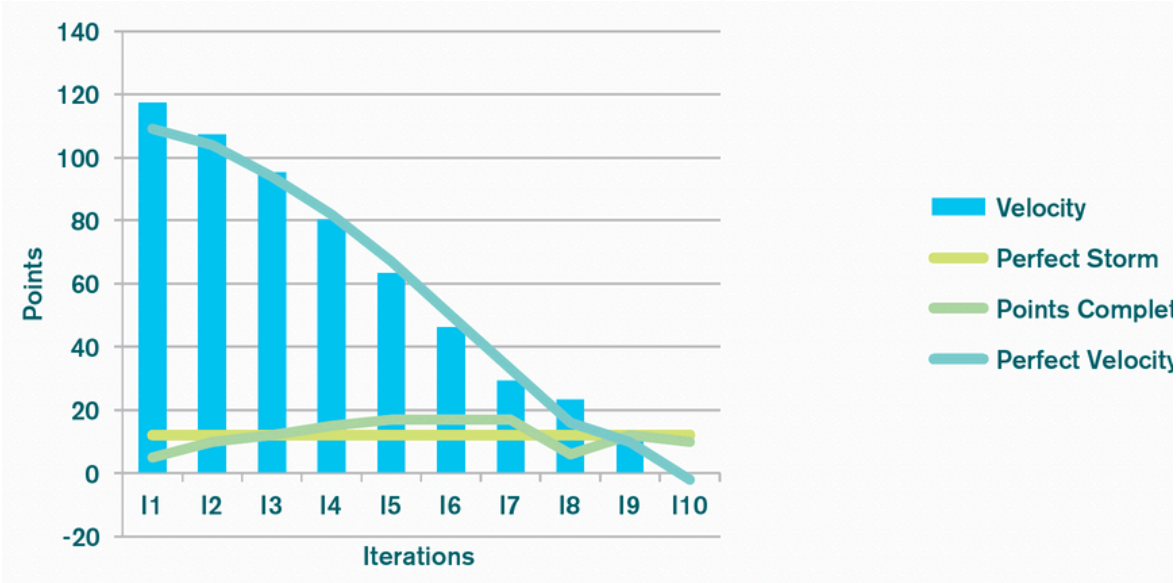
eBOOK

Starting out with disruptive technologies: Why a DevOps solution makes sense

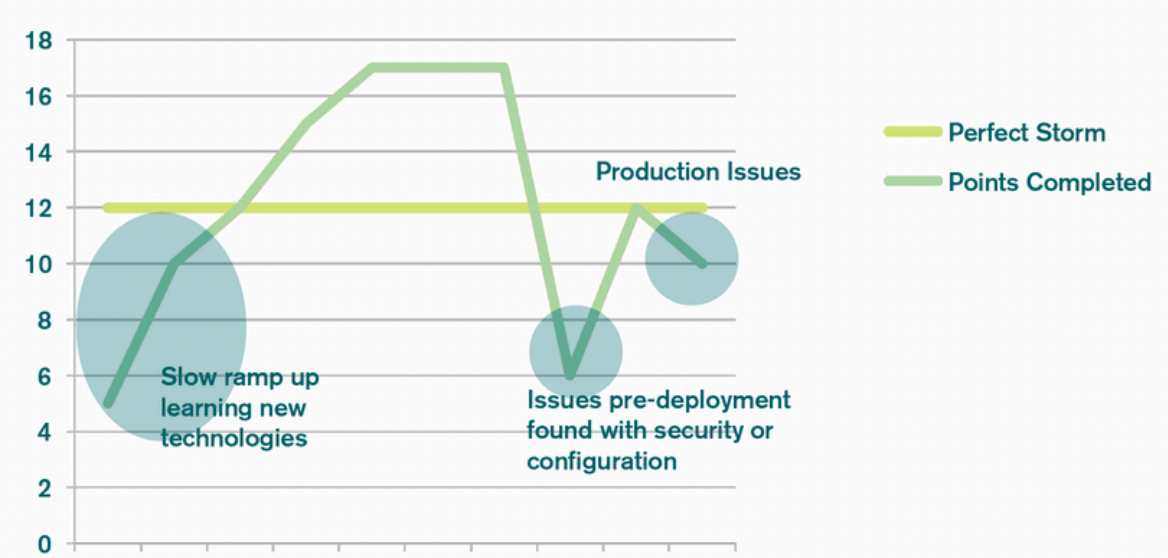
Solutions that constantly add new features, change processes and somehow simplify things as they get more complex, tend to have the highest success in the engagement and acquisition of customers. The companies that employ a team to do this often give that team the flexibility to perform their tasks the way they want to do it - and more often than not they choose a DevOps approach.

The cost of kick-off and that first production release
Technology decisions are often made at the project kick-off. Some of these choices will evolve and change over the duration of the project, while others will stay the same - and sometimes these decisions are regretted. Most of the decisions made take advantage of new technologies available to the team, technologies they want to dabble in and eventually become experts in.

It's these new technologies that help the solution be disruptive, moving away from the norm and finding ways to reduce overall costs and improve your time to market. But the delay these new technologies can bring could impact projects by at least two iterations, and slow the velocity down over all iterations. Gaps are often left open due to the inexperience of the team and this increases the risk of security holes and hardening of the solution. This could have a huge impact – not just on the project, but on the organisation as a whole.

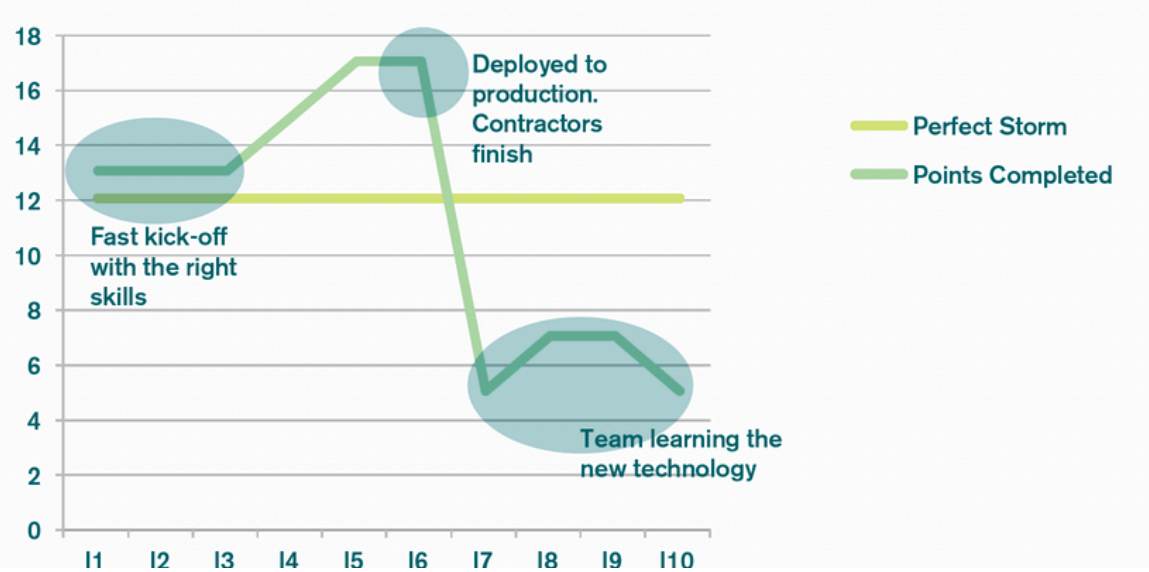


The above chart shows velocity – but what are the things that impact it, both positively and negatively? Let's now take a closer look at the important items in the chart, and find out more.



From this graph you can see where the slow points happen in a project that is being handled by an in-house team.
Slow start impacts velocity
Midway configuration or significant code changes impact velocity
Production issues impact velocity, as the team is taken away from project to resolve the problem.

To eliminate these issues, hiring in experienced contractors will improve the project's kick-off time and the velocity - for at least the period they are contracted. But as the transition of technology skills and a ramp up in required experience brings the velocity back down again, if left unchecked it can get to the point where the coding created may end up being dumped altogether.



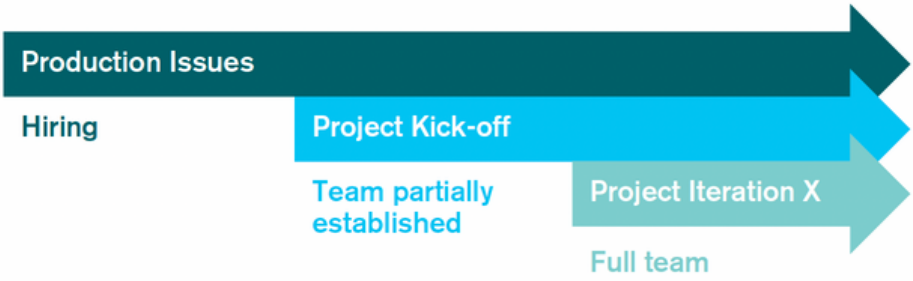
The above graph shows the impact points for a project where external contractors are hired in short-term, to get the project up and running.

- Kick-off is much faster
- Production release happens faster
- Team is held back trying to learn the new technology and manage production issues
- In some cases, new technology is dumped because “contractors did it”.

Both the above scenarios require your team to learn the new technologies fast and that moves them away from the building of the solution.

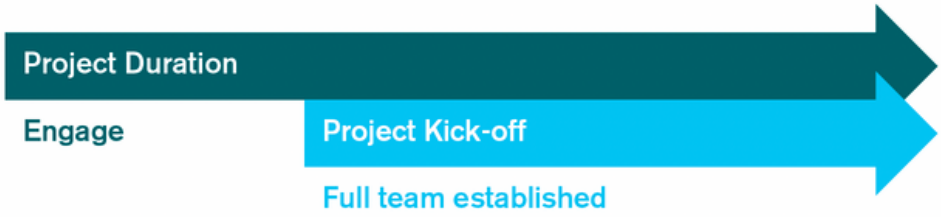
The exposure you get when going to production on your first release is also one that causes impact. When working with new technologies, it’s difficult to ensure that you have covered enough bases to build a robust production solution that can handle certain failures fast. Time needs to be spent on hardening the environments and in some cases this can lead to doubling your project iterations. Further more, as the team focuses on production issues, the velocity on new features drops dramatically, which then impacts on the release of these features.

Getting the expertise
Hiring in the right people – technically - is great, but what you are building is a product, not a tech company. The upfront costs and the delay in getting started as you find the right people could slow down any project by months. You will also need to get a wide range of skills to accommodate your technical breadth.



The alternative, that is gaining rapid traction worldwide, is to leverage a true DevOps company or group of companies. Companies that have a deep focus on DevOps have the experience, the pre-built configurations and the knowledge to get you started very quickly. A true DevOps company does not provide one time consulting services, they provide a managed service style agreement that provides assistance throughout the initial project, including post go-live.

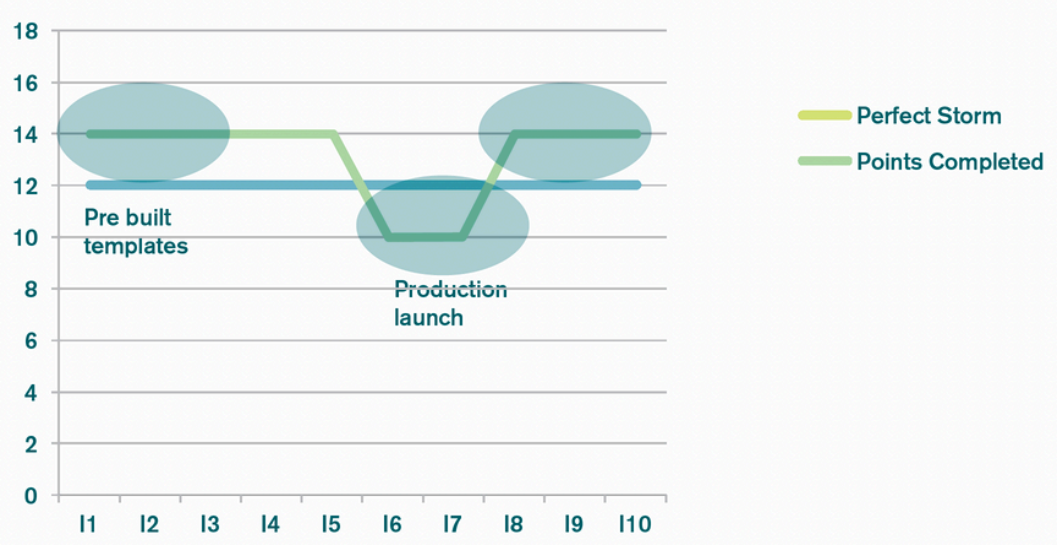
The benefits they should bring include the pre-defined processes that create a baseline for the services, new technology expertise (they often have used what you are trying to use before), and very fast ramp up and change times. A DevOps company should also assist in the ongoing support of the system. This gives you a velocity ramp up that starts much higher and helps you deliver the features much faster. It also limits the impact on your velocity during production issues.



Without this, any issues that arise tend to become the problem of the team that is trying to build the features.

Transition of skills is also a key factor to consider, and deciding when the right time is for your team to start owning all the new technology introduced. The impact on velocity here can be just as bad as during kick-off when your team ventures at it alone. And you need to consider the time it takes to take real ownership.

The longer the transition of skills takes, the more beneficial it is to the team, the velocity and the overall goals. Why? Because your team truly takes ownership rather than trying to hack their way through the challenges if it is only handed over.



The impact in the above graph showing the DevOps model is very clear.

- DevOps companies have the pre-built technology with the skills to get things moving faster. This is different to just hiring a contractor in.
- Production launch and issues may impact features but nowhere near as much as if your team had to completely re-focus. Production release happens much earlier for the features you take on.
- Your team is back to normal velocity much quicker

The models and project timelines
The models indicated above all offer varying levels of different results and in particular, going live is one of them. The shorter timelines for going live and the ability to ramp up again and again faster, definitely has an obvious cost benefit.

The key steps to success (technology focus)
The models indicated above all offer varying levels of different results and in particular, going live is one of them. The shorter timelines for going live and the ability to ramp up again and again faster, definitely has an obvious cost benefit.

- Start with the idea of what you are trying to achieve.
- Think about how you would do this with what you already know.
- Think about the development philosophy you want to take. Not talking about Agile vs Waterfall, but Domain Driven Design, Server-less design, integration patterns and even KISS principle.
- Find the items you are most comfortable with, i.e.: the database, and mark it for your first target. Why are you developing against a database? Is it really needed? Really? Why?
- Determine your system's logical components. That is, what makes up billing, what makes up a customer, what makes up "the widget".
- Which of these components do you really need to build? Is there a cloud based service that already does this for you? Can you get started with this cloud based service? Is it really going to give you an edge?

Once you have exhausted yourself with these, you can start thinking about the development cycle. That's right, forget about production because you wont be there for a while. The development cycles, the process, the SOE and the tools are the most important part of any product and project. It's true - without these in place how are you going to get started and keep things running and changing in production?

- The CI process and how you get things into a dev cycle is really important. Get this right and the team will work nicely together.
- The SOE is something that seems to be left out of many projects. This can potentially take up a couple days if dev configuration is not established for each team member. Look at Vagrant or Docker to get this right.
- Agree to a process, or at least to what is to be the start of the process. Check-in, tag, branch, fork, unit testing etc. - how are you going to handle tech blame?

BASE2 SERVICES

What is Velocity and how to measure it
Velocity is an extremely powerful method for accurately measuring the rate at which an Agile team delivers business value consistently. You start by simply adding points as estimates to each feature, story or requirement and sum up the total points delivered per iteration.

After a couple iterations, you can then start to use your historical data from the first two iterations, to more accurately create the points for the deliverables in each iteration. Don't try to use velocity from one project to calculate the velocity on another - it is meant to be localised. However, measuring success with velocity across multiple projects when the same teams operate on these projects is possible. This is how we established the metrics for this paper.

Assumptions made for this paper
The charts are designed to show accuracy and therefore the accumulation of results from multiple projects is fairly difficult. To build something more accurate we assumed the following:

- The requirements for each project were identical
- What was delivered at the end of the initial project was identical
- The issues in post release are identical.

What really works
In this paper we have described some of the ways that companies get started on disruptive technologies, and how the velocity impacts timelines. It's important to carefully consider the requirements of your project, the cost of development and how you can best maximise your team to disrupt.

Choosing a DevOps solution can help you get the best possible results in the shortest time and lessen the impacts that cause delays – from kick-off, to production and beyond.

Looking for an expert?
If you are considering migrating to cloud technology, have experienced issues in the cloud or have concerns regarding the best solution for you, we can help. We take the complexity out of cloud technology, regardless of your experience or capabilities.

Our team is one of the industry's best and readily available to assist with every stage of your integration and cloud adoption process. Our "Disruptive Workshops" ensure that you receive the advice you need to move your business forward.

Get in touch

BASE2 Services is a global leader in Cloud Delivery, Operations and Management. Specialising in DevOps, AWS and cloud-native computing.

We enable organisations to accelerate innovation through automated, highly secure, repeatable and scalable cloud-based solutions. Contact us today to find out how we can help you.

USA

11801 Domain Blvd
3rd Floor
Austin TX 78758
+1 646 586 9485

AUSTRALIA

Level 3
44 Clifton St
Prahran VIC 3181
1300 713 559

GERMANY

4. OG
Potsdamer Str. 182
10783 Berlin
+49 30 2000 5370

info@base2services.com

base2services.com.au