

This is part provocation, part therapy. Don't take it personally—it's in good faith!

///

Marcel British | Beautiful Abstraction
Digital consultant, product manager and facilitator

Marcel Britsch is an independent Digital Consultant, Product Manager and Agile Transformation specialist.

He has been living and working in London for over 20 years. He has worked with creatively and technically focused agencies, consultancies and clients across retail, education, conservation, automotive, finance, healthcare and energy.

He helps organisations build solid products and services in a sustainable way by facilitation, pairing, coaching or hands-on product management.

He believes that project success is strongly linked to happy teams, value-focused decision-making and fast feedback cycles. He is passionate about finding the best tools and techniques to optimise team culture, ways of working and solution design. He considers projects that follow classic waterfall / big-design-up-front practices to be too likely doomed to go anywhere near them, but loves to help organisations build products and transform in incremental evolutionary fashion or move towards this approach.

Outside of work he is interested in SciFi and comic books, Theravada Buddhist meditation and number theory.

He is a regular speaker at conferences, teaches at Cambridge University, blogs at www.thedigitalbusinessanalyst.com, hosts the <https://www.theburnup.com> podcast about 'all things agile', and can be found at <https://www.beautifulabstraction.com>.

—

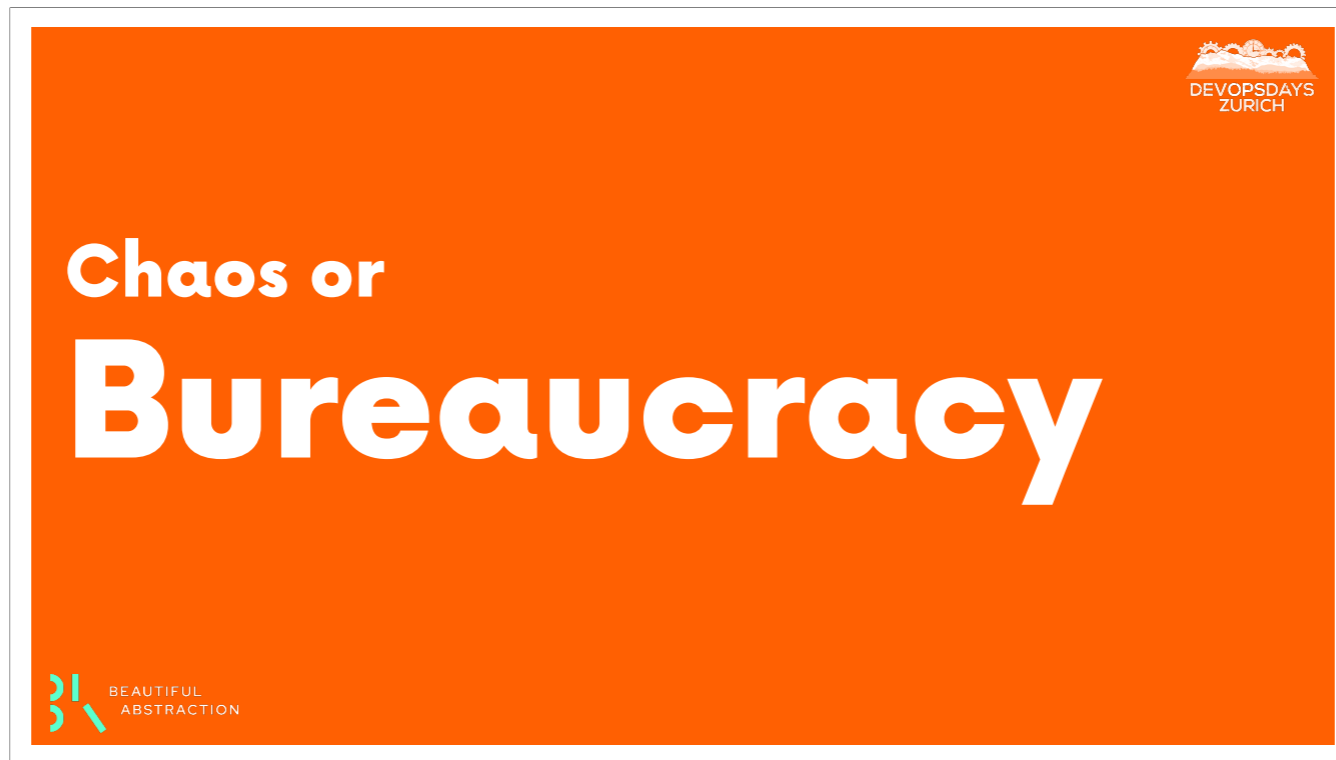
Web: <https://www.beautifulabstraction.com>

Blog: www.thedigitalbusinessanalyst.com

Podcast: <https://www.theburnup.com>

Profile: <https://www.linkedin.com/in/marcelbritsch>

Social: <https://mastodon.social/@marcelbritsch>



When I started my career, software delivery was either unmanaged and chaotic or overly bureaucratic, inefficient and slow.



Then we got CI/CD, automation, infrastructure as code, evolutionary architecture! The promise? Smooth, speedy, pain-free delivery... right, right?



From where I'm standing more often than not, that's not the case: startups and enterprises alike: teams are stuck fixing, fiddling with their platform rather than delivering business or end user value. Just that last thing, I'm being told, then we have flow. But there is always one more last thing that needs doing...



Now: I've seen great delivery—like the UK tax office, deploying 30–40 times a day.



But far too often I see startups burning their funding or enterprises distracting teams by building overly ambitious platforms at times when they should be focusing on delivering actual to end users and the business.



So why does it still go wrong? I want to share some antipatterns that my DevOps colleagues and I have observed.



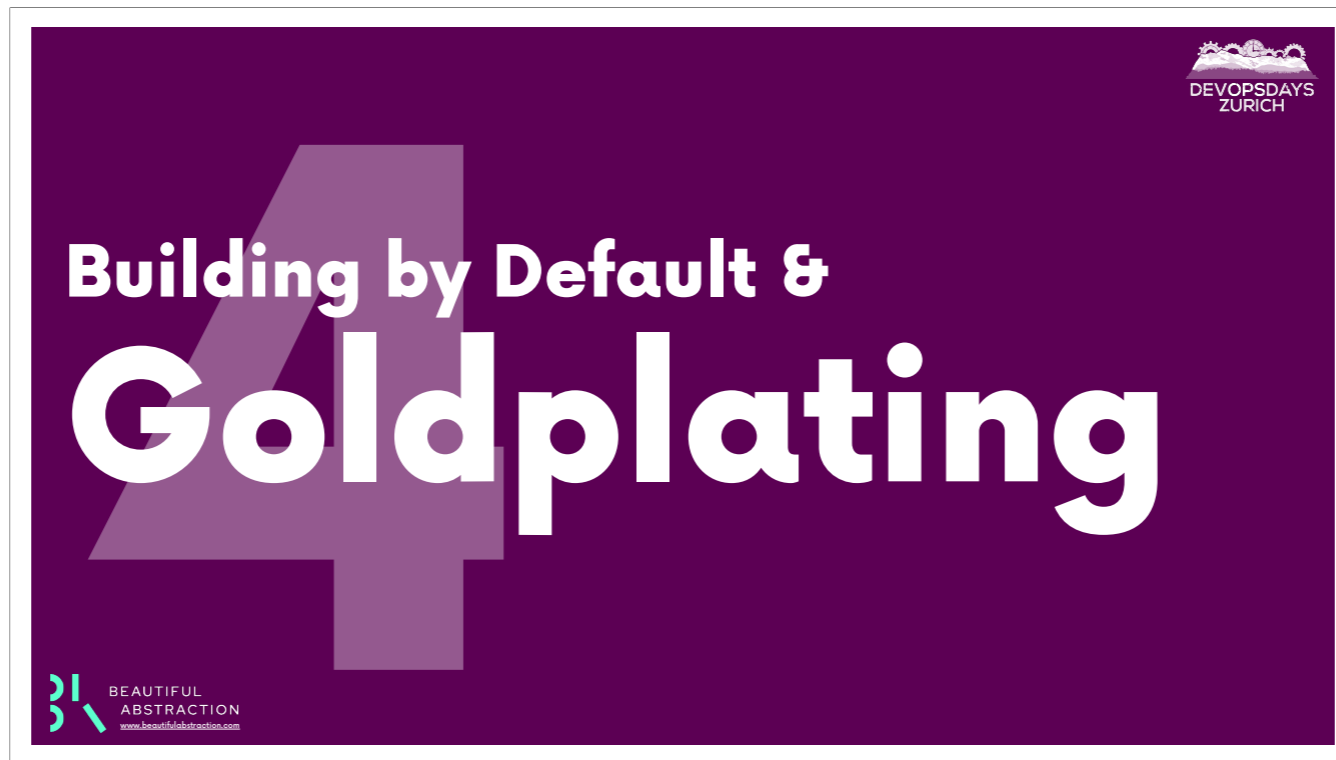
We opt for Kubernetes, Microservices or Lambdas without considering if they're truly needed, often because they're trendy or "Google uses them." I've seen companies miss contractual deadlines, unable to deploy core features, while building for a level of cloud agnosticism they didn't need.



We chose esoteric frameworks or languages, implement blockchain based logging, or pursue event-driven architectures— telling us these provide benefits, while in fact we want to explore these technologies and have them on our CV.



We fixate on small details or areas where we are comfortable —like naming conventions or tool choices—because this procrastination is easier than deciding on critical architectural choices or solving actual domain problems.



We default to building (because that's what we do) when we should be buying. We justify this by pointing to the future, overlooking that such adding of complexity or premature optimisation is not only a waste of time and effort now, but that the added complexity will be an issue now and bite us in the future.

Misunderstood goals are the real
Reason
we keep messing up

So why do we see these anti-patterns? I think it all comes down to misunderstood or misaligned goals and incentives.

All we want is

Features

**in the hands of users, sustainably
at pace & with feedback loops**

Turning ideas in people's minds into features in the hands of users. And do so sufficiently reliably, at pace, with tight feedback loops. Everything else is just a means to this end.

DEVOPSDAYS
ZURICH

At minimum our pipeline is an
Enabler

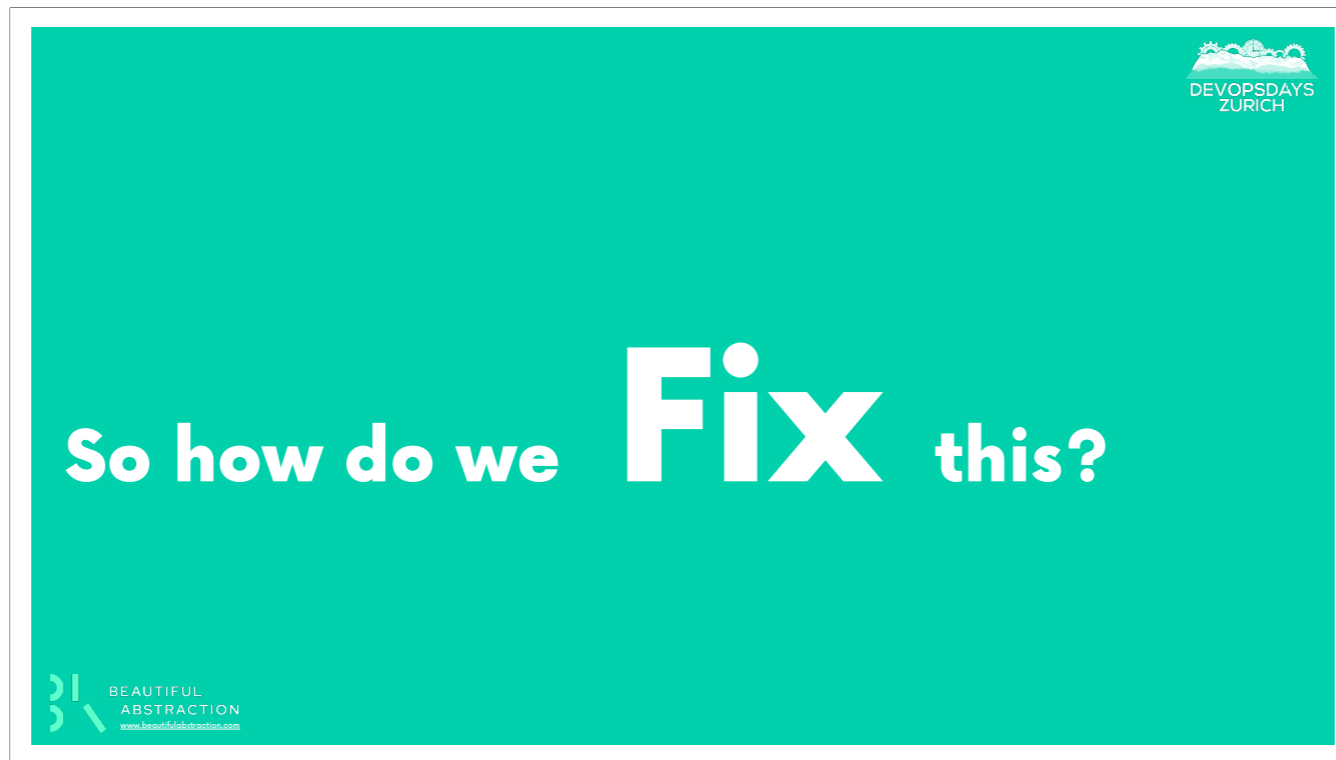
BEAUTIFUL
ABSTRACTION
www.beautifulabstraction.com

At the bare minimum, the pipeline—tooling, infra, architecture, process—is an enabler, not the goal. If it slows delivery, it's failing its purpose.

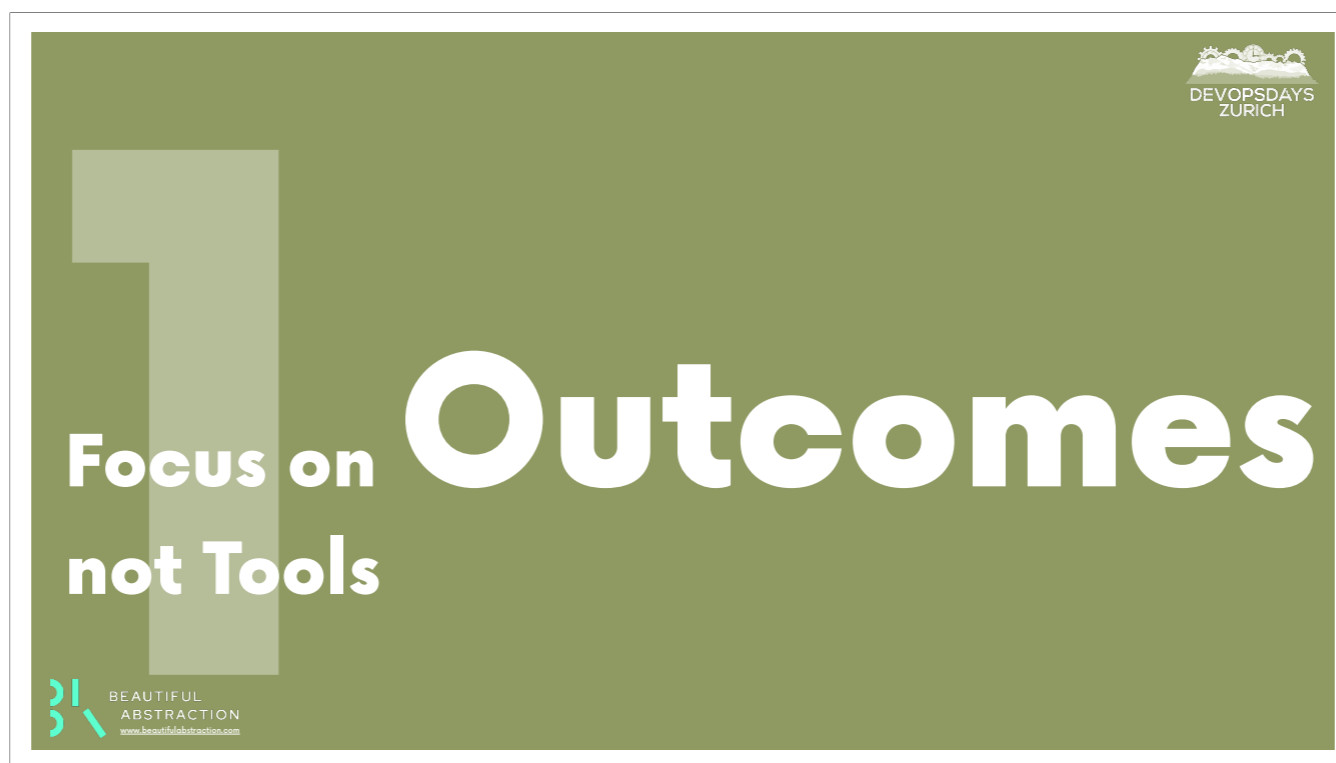


Done exceedingly well, our platform / pipeline becomes competitive advantage. Like in the case of one of my MedTech clients who were able to automate regulatory concerns as part of their pipeline, cutting regulatory deployment cycles from six months to two days, and consequently having one of the industry's most successful IPOs.

I have outlined why your pipeline matters as strategic asset, that it needs to be managed like a product and this MedTech example in great detail here: <https://www.beautifulabstraction.com/productthinking-devops/>



One colleague said to me: to know great CI/CD you'll have seen it. Point being, it's hard and possibly counter-intuitive. Here are three suggestions to help:



The platform is not a technical playground. It is all about supporting value delivery. This means building platforms, but the right one.



One size doesn't fit all—find your organisation's sweet spot in terms of what they need from their platform and what they can support.



Keep it lean. Focus on flow. Build the thinnest sufficiently reliable platform. Then evolve iteratively and incrementally as you would any other product.

Again, there is much more detail about this here: <https://www.beautifulabstraction.com/productthinking-devops/>



DevOps is not the goal
Value delivery is



Marcel Britsch
Product consultant and facilitator

Web: www.beautifulabstraction.com
Blog: www.thedigitalbusinessanalyst.com

If you're shipping features fast, learning continuously, and delivering real value sufficiently reliably—you're doing DevOps right.

If, on the other hand your PM is whining, shouting and bitching at you, maybe it's time to review your approach.

More here: <https://www.beautifulabstraction.com/productthinking-devops/>

Thank you - please do get in touch...

////

Marcel British | Beautiful Abstraction

Digital consultant, product manager and facilitator

Marcel Britsch is an independent Digital Consultant, Product Manager and Agile Transformation specialist.

He has been living and working in London for over 20 years. He has worked with creatively and technically focused agencies, consultancies and clients across retail, education, conservation, automotive, finance, healthcare and energy.

He helps organisations build solid products and services in a sustainable way by facilitation, pairing, coaching or hands-on product management.

He believes that project success is strongly linked to happy teams, value-focused decision-making and fast feedback cycles. He is passionate about finding the best tools and techniques to optimise team culture, ways of working and solution design. He considers projects that follow classic waterfall / big-design-up-front practices to be too likely doomed to go anywhere near them, but loves to help organisations build products and transform in incremental evolutionary fashion or move towards this approach.

Outside of work he is interested in SciFi and comic books, Theravada Buddhist meditation and number theory.

He is a regular speaker at conferences, teaches at Cambridge University, blogs at www.thedigitalbusinessanalyst.com, hosts the <https://www.theburnup.com> podcast about 'all things agile', and can be found at <https://www.beautifulabstraction.com>.

Web: <https://www.beautifulabstraction.com>

Blog: www.thedigitalbusinessanalyst.com

Podcast: <https://www.theburnup.com>

Profile: <https://www.linkedin.com/in/marcelbritsch>

Social: <https://mastodon.social/@marcelbritsch>