SESSION 301
Wednesday, November 2, 3:00 PM - 4:00 PM
Track: DevOps and Agile

Love DevOps? Wait 'Til You Meet Site Reliability Engineering!

Patrick Hill
Site Reliability Engineering Lead,Atlassian
patrick@atlassian.com

Session Description

Reliability, arguably the most important feature of any service, can suffer when development and operations team objectives are set up in opposition to one another. Developers want to get their features out to real users rapidly and frequently, while operation's team objectives are centered around and measured against availability metrics of the services themselves. To improve stability and drive down incidents, Site Reliability Engineering (SRE) facilitates collaboration and alignment with development teams, increasing the safety and frequency of software releases. In this session, you'll learn how to creatively overcome many of the limitations of traditional service operations and take the drama out of DevOps. *(Experience Level: Intermediate)*

Speaker Background

Patrick Hill is the Site Reliability Engineering (SRE) team lead at Atlassian, and recently transferred from Sydney to Austin. (G'day, y'all!) In his free time, he enjoys taking his beard from "distinguished professor" to "lumberjack" and back again. Find him on Twitter @topofthehill.


FUSION 16
THE NEXT GENERATION OF SERVICE MANAGEMENT
Love DevOps? Wait 'til you meet SRE!
Patrick Hill – Atlassian - @topofthehill
PRODUCED BY: itSMF USA HDI
#SMFUSION

HOW WE GOT TO WHERE WE ARE

- I’m Patrick Hill, SRE Team Lead at Atlassian
- Our company was founded in Australia, and has primarily thrived off products that are able to sell themselves, in part due to our geographic distance from the Software market
- For most of our company’s history we’ve been focusing on downloadable software that you can run on your own infrastructure!

HOW WE GOT TO WHERE WE ARE

Build great software…

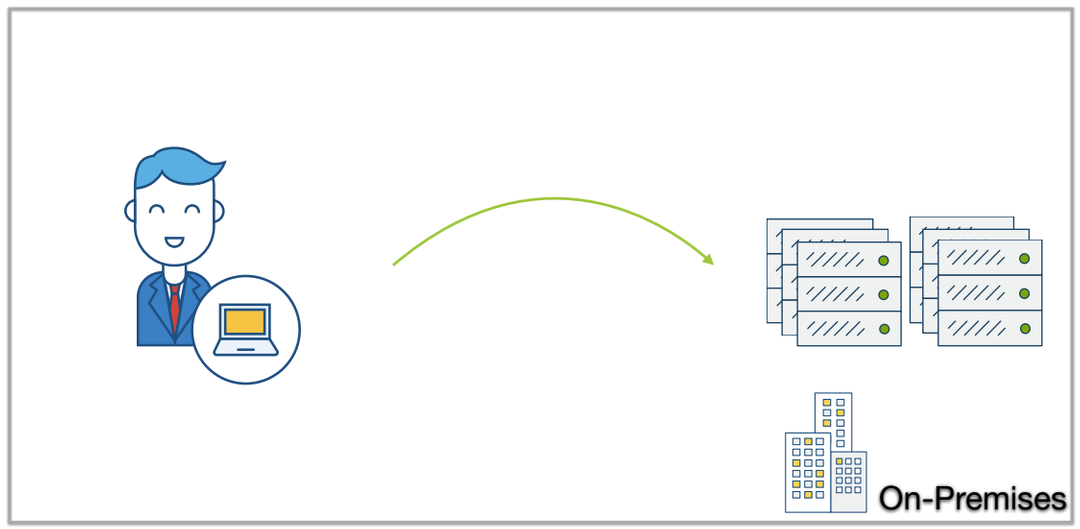
… that sells itself!

- I'm Patrick Hill, SRE Team Lead at Atlassian
- Our company was founded in Australia, and has primarily thrived off products that are able to sell themselves, in part due to our geographic distance from the Software market
- For most of our company's history we've been focusing on downloadable software that you can run on your own infrastructure!

HOW WE GOT TO WHERE WE ARE

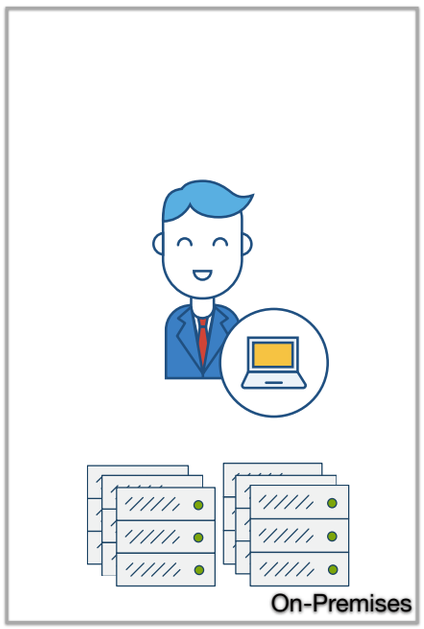
- For a long time this worked, customer ran themselves, Atlassian shipped features and bug fixes
- We and customers grew, customers were looking for a hosted model
- So we built Atlassian Cloud. This has gone through several iterations as we started off with custom managed hosting solutions to a more native SaaS model
- As our customer base has grown, more and more customers have had an appetite for Cloud services
- New acquisitions and services

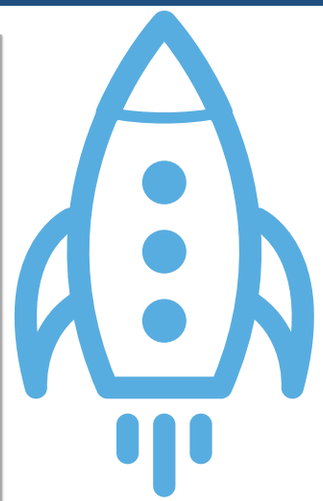
HOW WE GOT TO WHERE WE ARE



- For a long time this worked, customer ran themselves, Atlassian shipped features and bug fixes
- We and customers grew, customers were looking for a hosted model
- So we built Atlassian Cloud. This has gone through several iterations as we started off with custom managed hosting solutions to a more native SaaS model
- As our customer base has grown, more and more customers have had an appetite for Cloud services
- New acquisitions and services

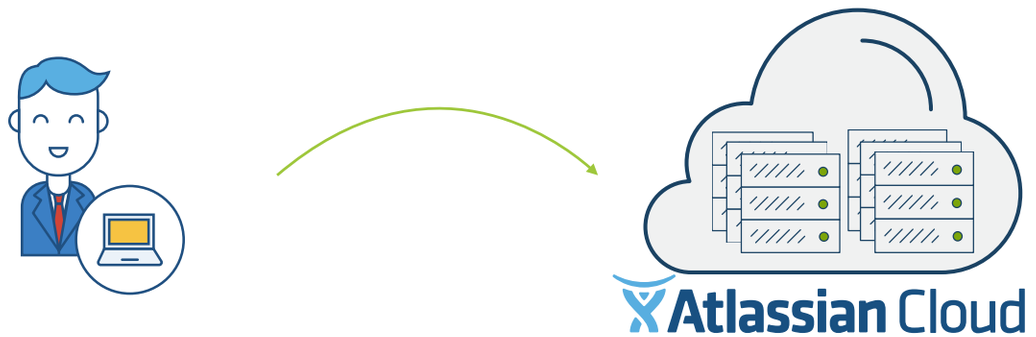
HOW WE GOT TO WHERE WE ARE





- For a long time this worked, customer ran themselves, Atlassian shipped features and bug fixes
- We and customers grew, customers were looking for a hosted model
- So we built Atlassian Cloud. This has gone through several iterations as we started off with custom managed hosting solutions to a more native SaaS model
- As our customer base has grown, more and more customers have had an appetite for Cloud services
- New acquisitions and services

HOW WE GOT TO WHERE WE ARE



- For a long time this worked, customer ran themselves, Atlassian shipped features and bug fixes
- We and customers grew, customers were looking for a hosted model
- So we built Atlassian Cloud. This has gone through several iterations as we started off with custom managed hosting solutions to a more native SaaS model
- As our customer base has grown, more and more customers have had an appetite for Cloud services
- New acquisitions and services

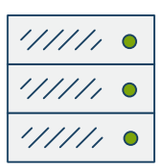
HOW WE GOT TO WHERE WE ARE

- For a long time this worked, customer ran themselves, Atlassian shipped features and bug fixes
- We and customers grew, customers were looking for a hosted model
- So we built Atlassian Cloud. This has gone through several iterations as we started off with custom managed hosting solutions to a more native SaaS model
- As our customer base has grown, more and more customers have had an appetite for Cloud services
- New acquisitions and services

A little story about growth

- Now in a situation where dev shipped products, features to cloud + internal
- **(*)**
- Service Operations was a result
- **(*)**
- Worked pretty well to start with, people shipped stuff, occasionally something went wrong

A little story about growth

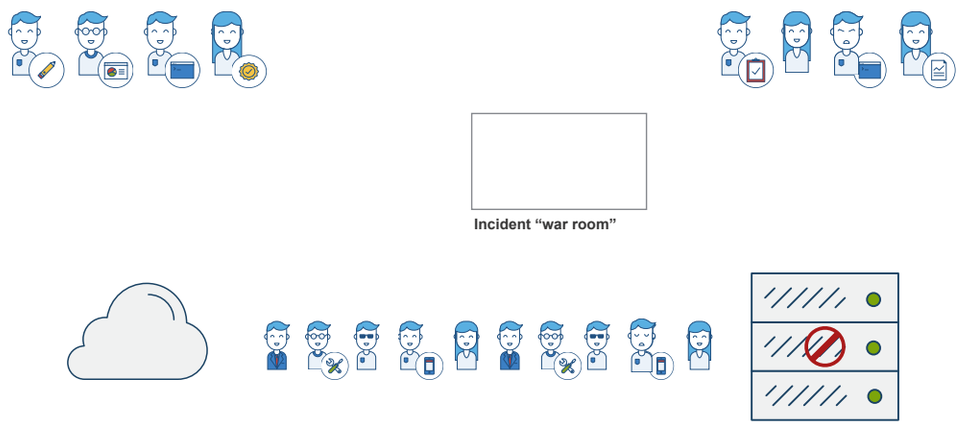


- Now in a situation where dev shipped products, features to cloud + internal
- **(*)**
- Service Operations was a result
- **(*)**
- Worked pretty well to start with, people shipped stuff, occasionally something went wrong

A little story about growth

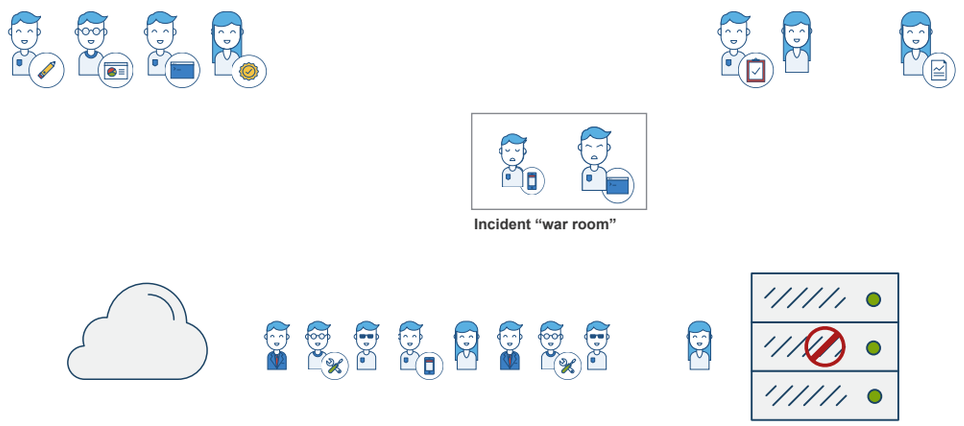
- Now in a situation where dev shipped products, features to cloud + internal
- **(*)**
- Service Operations was a result
- **(*)**
- Worked pretty well to start with, people shipped stuff, occasionally something went wrong

A little story about growth



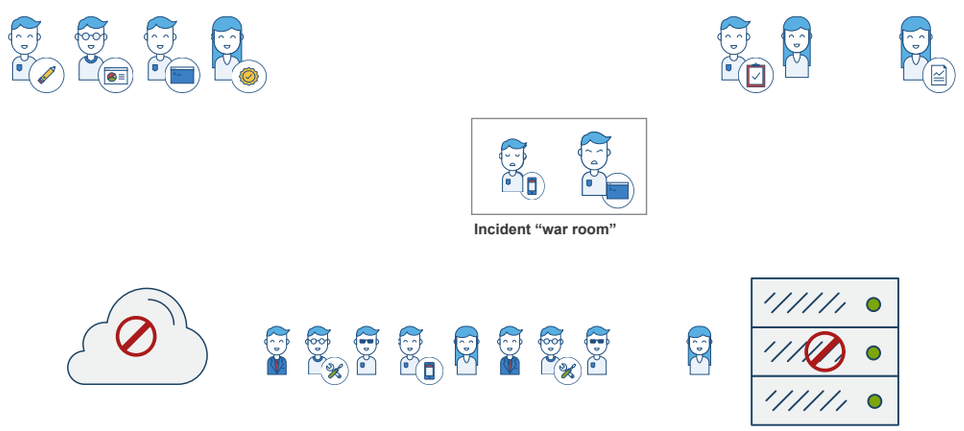
- Apply company-wide incident management, SOPS and Devs together
- **(*)**
- At the same time, other teams shipping features - funnily enough, more goes wrong
- Get more people together to fix the problem
- **(*)**
- Patched/backed out
- Big wins from centralisation - consistent IM, company comms, PM, reporting

A little story about growth



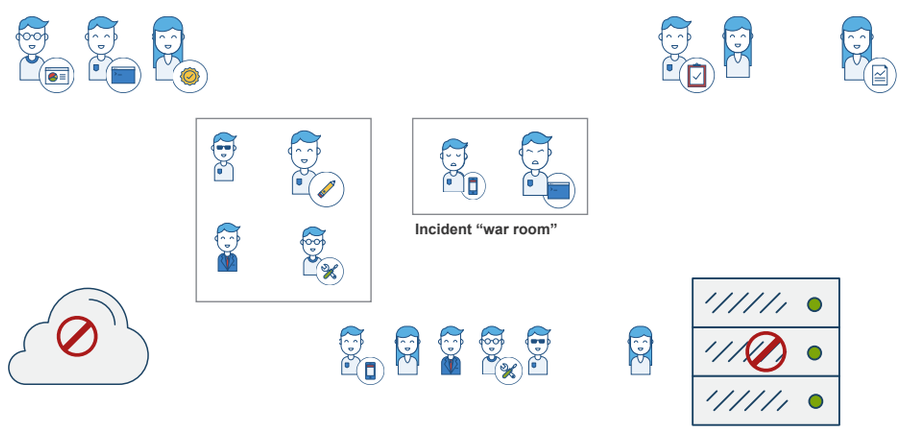
- Apply company-wide incident management, SOPS and Devs together
- (*)
- At the same time, other teams shipping features - funnily enough, more goes wrong
- Get more people together to fix the problem
- (*)
- Patched/backed out
- Big wins from centralisation - consistent IM, company comms, PM, reporting

A little story about growth



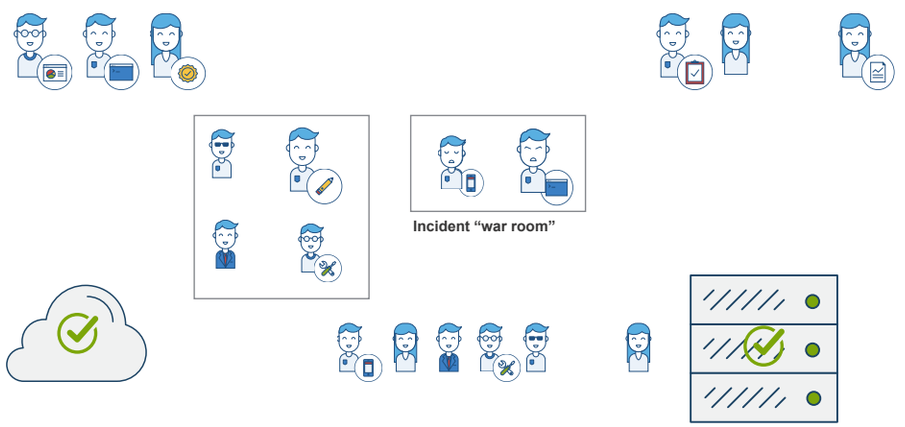
- Apply company-wide incident management, SOPS and Devs together
- (*)
- At the same time, other teams shipping features - funnily enough, more goes wrong
- Get more people together to fix the problem
- (*)
- Patched/backed out
- Big wins from centralisation - consistent IM, company comms, PM, reporting

A little story about growth



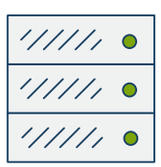
- Apply company-wide incident management, SOPS and Devs together
- **(*)**
- At the same time, other teams shipping features - funnily enough, more goes wrong
- Get more people together to fix the problem
- **(*)**
- Patched/backed out
- Big wins from centralisation - consistent IM, company comms, PM, reporting

A little story about growth



- Apply company-wide incident management, SOPS and Devs together
- **(*)**
- At the same time, other teams shipping features - funnily enough, more goes wrong
- Get more people together to fix the problem
- **(*)**
- Patched/backed out
- Big wins from centralisation - consistent IM, company comms, PM, reporting

A little story about growth

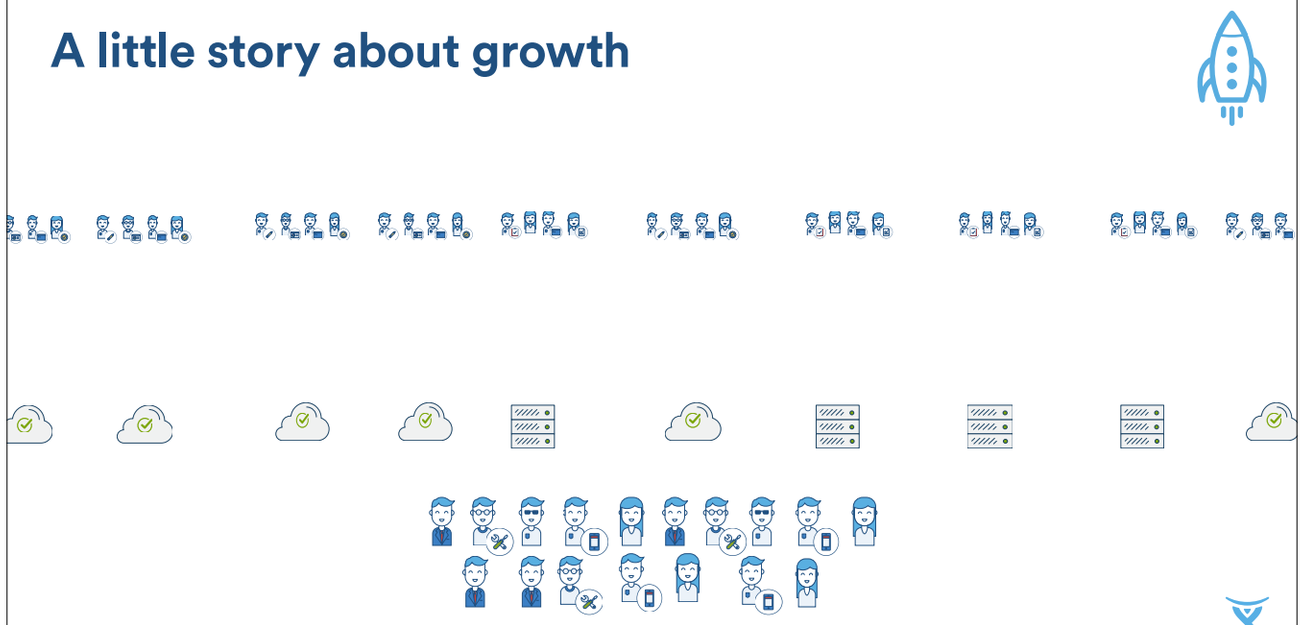


A little story about growth

A little story about growth

- Fast Forward a few years - more services, runbooks
- SOPS grew with the company

A little story about growth



- Fast Forward a few years - more services, runbooks
- SOPS grew with the company

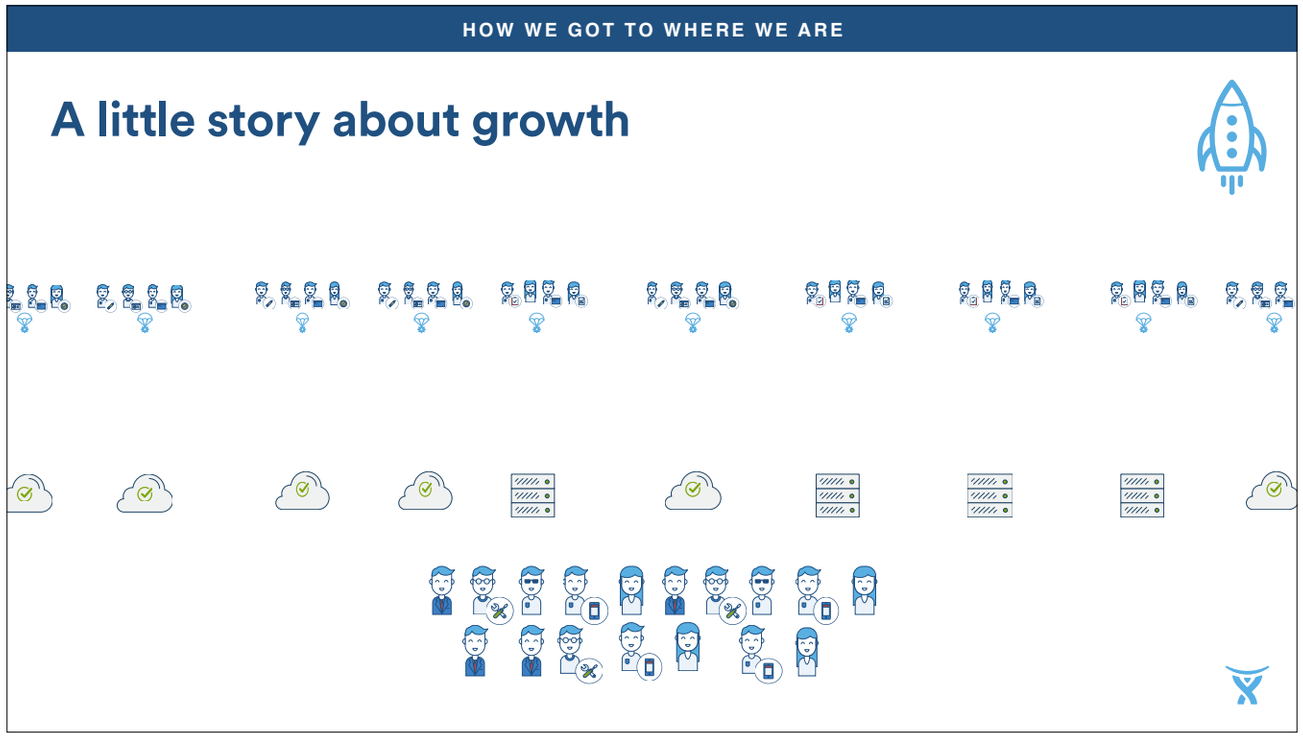


Image: fair use under http://www.insidecounsel.com/2013/06/21/technology-internet-memes-pose-legal-questions

- Consistent process was good, Engineers were less effective, not solving problems
- Only action to restart
- Relationship between dev and ops was changing
- Whenever we found ourselves in a situation like this…
- **(*)**
- We often found the response was this…
- **(*)**
- Fundamental problem with the way we were managing our services and the way we were engaging with our development teams.

Image: fair use under http://www.insidecounsel.com/2013/06/21/technology-internet-memes-pose-legal-questions

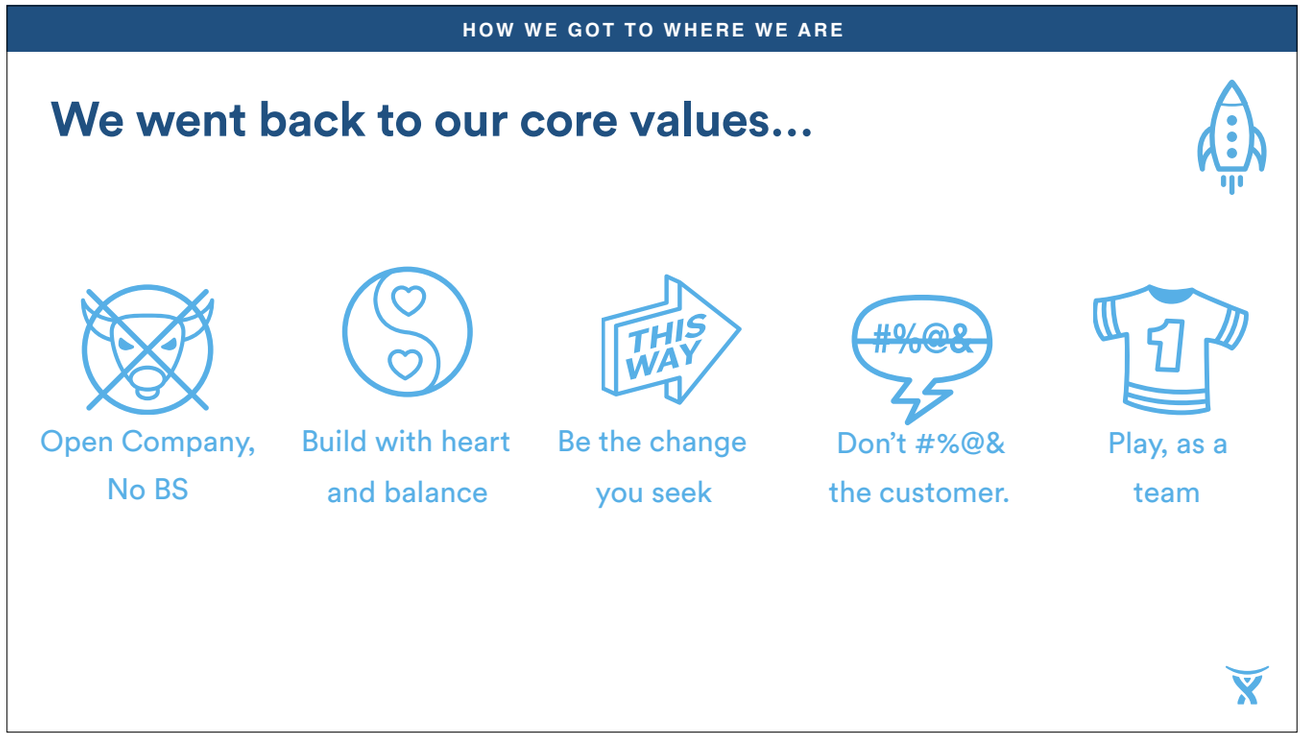
- Consistent process was good, Engineers were less effective, not solving problems
- Only action to restart
- Relationship between dev and ops was changing
- Whenever we found ourselves in a situation like this…
- **(*)**
- We often found the response was this…
- **(*)**
- Fundamental problem with the way we were managing our services and the way we were engaging with our development teams.

Image: fair use under http://www.insidecounsel.com/2013/06/21/technology-internet-memes-pose-legal-questions

- Consistent process was good, Engineers were less effective, not solving problems
- Only action to restart
- Relationship between dev and ops was changing
- Whenever we found ourselves in a situation like this…
- **(*)**
- We often found the response was this…
- **(*)**
- Fundamental problem with the way we were managing our services and the way we were engaging with our development teams.

We went back to our core values...

- Atlassian has 5 company values. This company lives the values - people talk about them
- We talked to Engineers, we weren't living our values. People weren't doing the best work of their lives.
- Yes 24hr coverage, service restored
- What about Capacity, Risk, DR, Reliability?
- Not as focused on reliability
- SOPS not building with heart and balance
- Not playing as a team - disconnect between SOPS and Dev
- In short - we weren't going anywhere
- **(*)**
- So we decided to do something about it



- Atlassian has 5 company values. This company lives the values - people talk about them
- We talked to Engineers, we weren’t living our values. People weren’t doing the best work of their lives.
- Yes 24hr coverage, service restored
- What about Capacity, Risk, DR, Reliability?
- Not as focused on reliability
- SOPS not building with heart and balance
- Not playing as a team - disconnect between SOPS and Dev
- In short - we weren’t going anywhere
- **(*)**
- So we decided to do something about it

We went back to our core values...

Be the change

you seek

- Atlassian has 5 company values. This company lives the values - people talk about them
- We talked to Engineers, we weren't living our values. People weren't doing the best work of their lives.
- Yes 24hr coverage, service restored
- What about Capacity, Risk, DR, Reliability?
- Not as focused on reliability
- SOPS not building with heart and balance
- Not playing as a team - disconnect between SOPS and Dev
- In short - we weren't going anywhere
- **(*)**
- So we decided to do something about it

- https://static.pexels.com/photos/6754/road-nature-lines-country.jpg - google says it's fair use
- So we went searching
- Traditional Ops, investment
- DevOps
- Something different - Engineers to code, build tools, focus on Reliability

We Went Searching

- Looked at some of the big service providers
- 15yrs vs startups
- Common focus on reliability
- “Availability”, “Disaster Recovery”, “Capacity Planning”, “Release Management”, “Solve problems with code”
- What we ultimately found was SRE

We Went Searching

Google

Dropbox

twitter

facebook

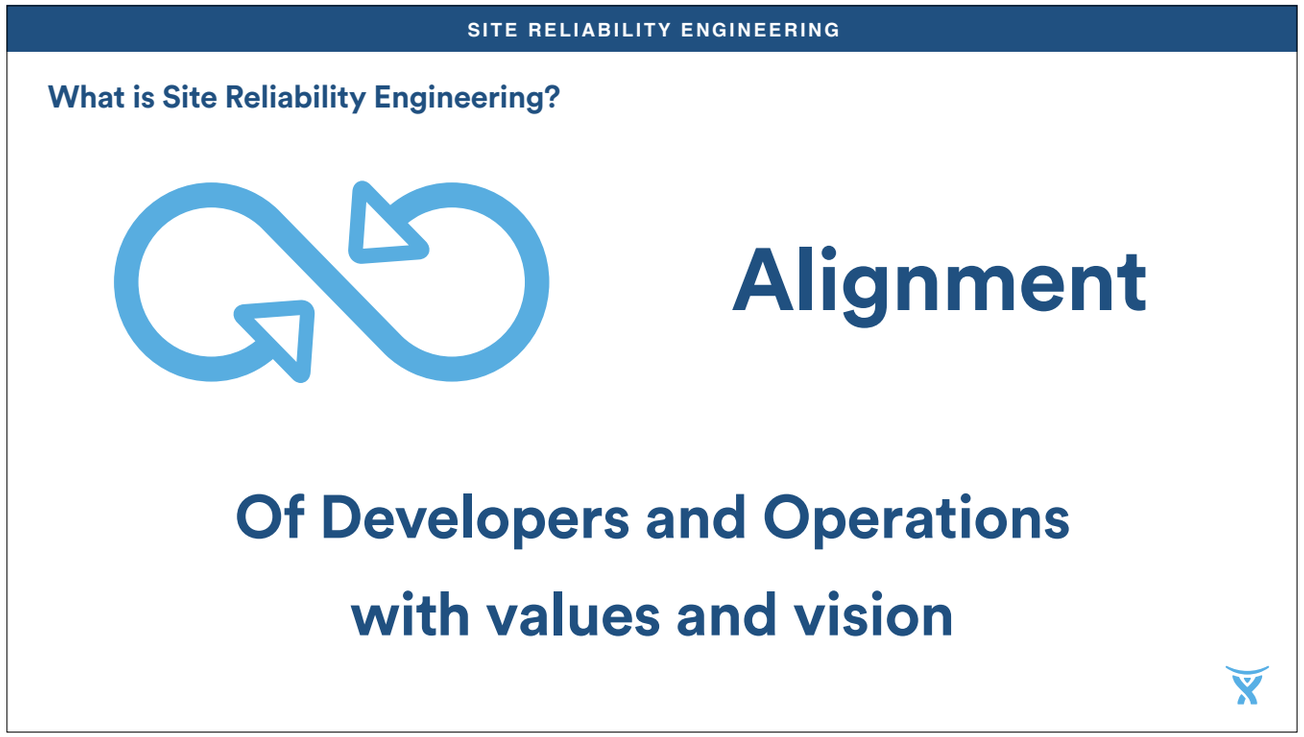
LinkedIn

Etsy

- Looked at some of the big service providers
- 15yrs vs startups
- Common focus on reliability
- “Availability”, “Disaster Recovery”, “Capacity Planning”, “Release Management”, “Solve problems with code”
- What we ultimately found was SRE

SITE RELIABILITY ENGINEERING

Site Reliability Engineering is a methodology for managing services that focuses on the alignment of Developers and Operations with company values and vision.



Site Reliability Engineering is a methodology for managing services that focuses on the alignment of Developers and Operations with company values and vision.

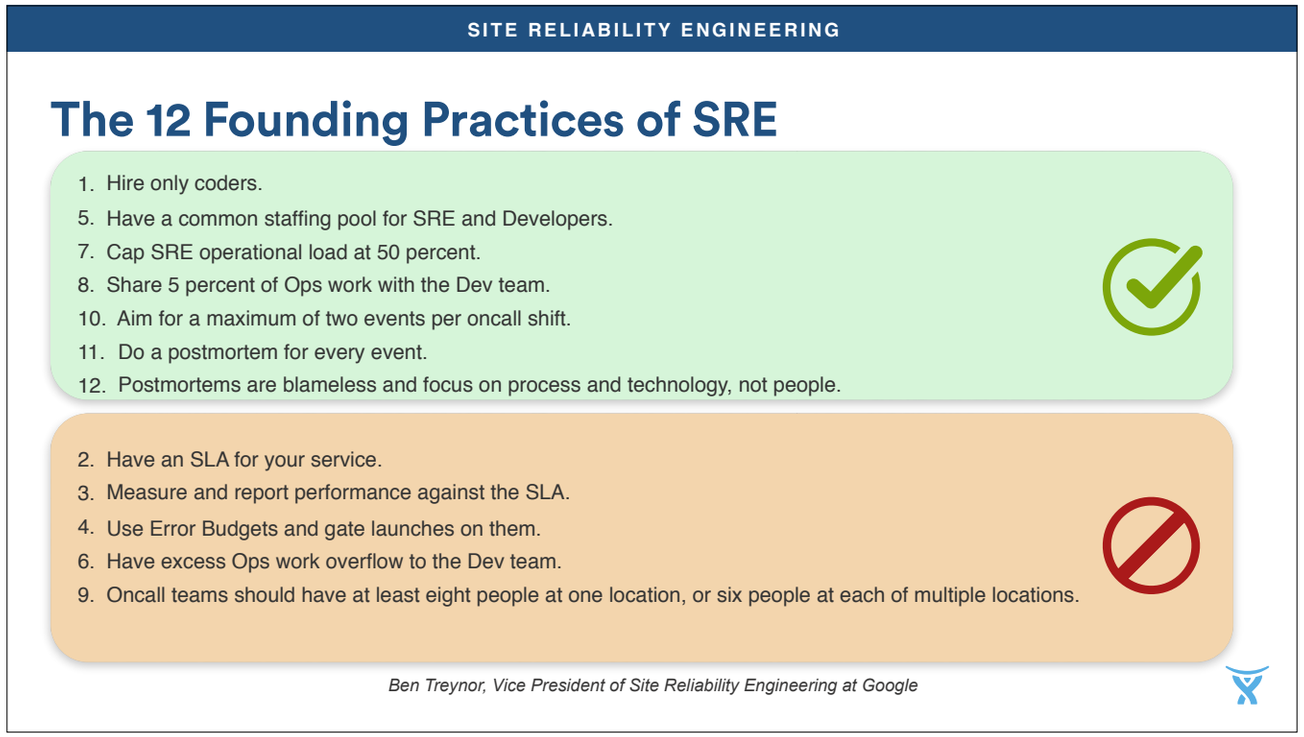
SITE RELIABILITY ENGINEERING

The 12 Founding Practices of SRE

1. Hire only coders.
2. Have an SLA for your service.
3. Measure and report performance against the SLA.
4. Use Error Budgets and gate launches on them.
5. Have a common staffing pool for SRE and Developers.
6. Have excess Ops work overflow to the Dev team.
7. Cap SRE operational load at 50 percent.
8. Share 5 percent of Ops work with the Dev team.
9. Oncall teams should have at least eight people at one location, or six people at each of multiple locations.
10. Aim for a maximum of two events per oncall shift.
11. Do a postmortem for every event.
12. Postmortems are blameless and focus on process and technology, not people.

Ben Treynor, Vice President of Site Reliability Engineering at Google

- Ben Treynor @ Google - 12 founding Practises
- Got excited - Devs doing Ops work - shared some DevOps Principles
- Organisational change
- **(*)**
- Picked a few, quick wins, focus areas
- Long term commitment to all
- All SREs can code - dev hiring pool
- Share the Ops load with Dev
- Reactive work to 50%
- PIR Focus



- Ben Treynor @ Google - 12 founding Practises
- Got excited - Devs doing Ops work - shared some DevOps Principles
- Organisational change
- **(*)**
- Picked a few, quick wins, focus areas
- Long term commitment to all
- All SREs can code - dev hiring pool
- Share the Ops load with Dev
- Reactive work to 50%
- PIR Focus

EXPERIMENT

- SRE worth a try
- Safe place to try it out
- **(*)**
- Had a look at landscape
- Flagship Products, newer services, infrastructure, purchasing
- What are we trying to achieve? improve things, Reliability, ALL IN!
- **(*)**
- Our first SRE team manages our Purchasing and Order Management Services – without which, customers can’t buy our products. It’s a good thing it worked out well!

EXPERIMENT

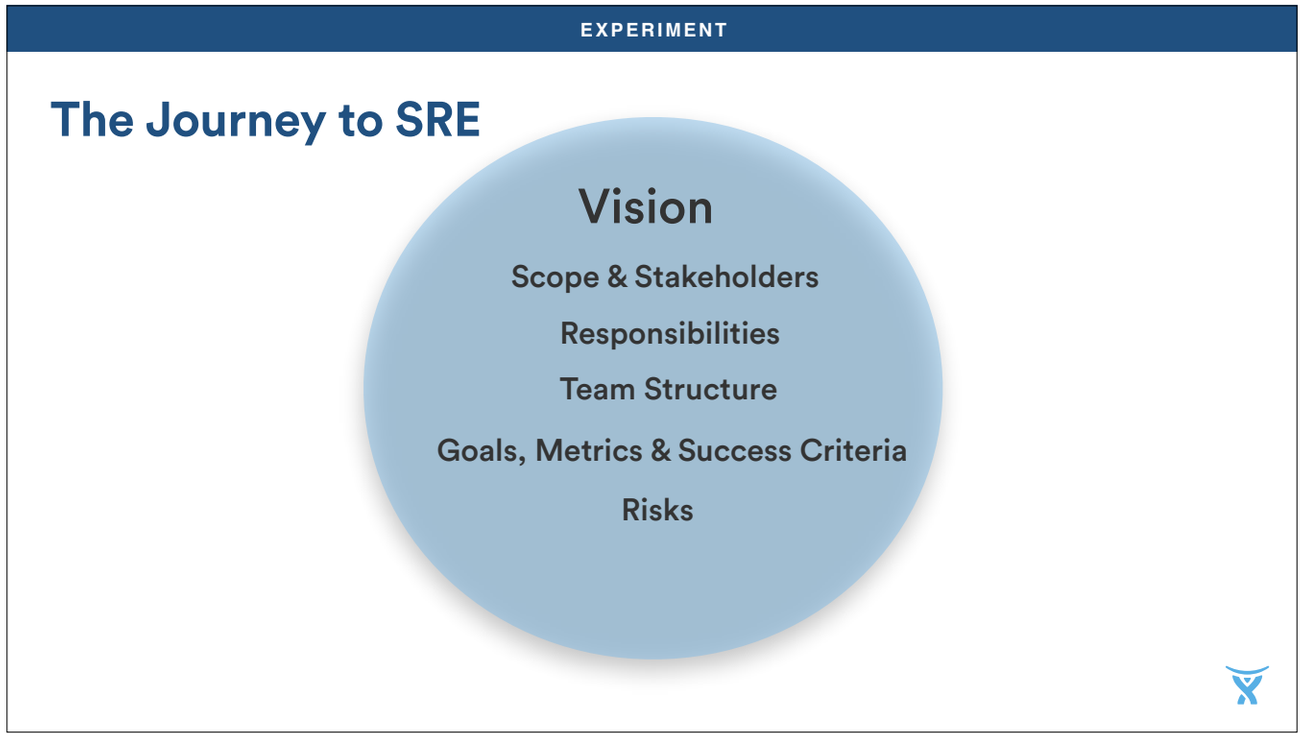
SRE?
Let’s give it a try.

- SRE worth a try
- Safe place to try it out
- **(*)**
- Had a look at landscape
- Flagship Products, newer services, infrastructure, purchasing
- What are we trying to achieve? improve things, Reliability, ALL IN!
- **(*)**
- Our first SRE team manages our Purchasing and Order Management Services – without which, customers can’t buy our products. It’s a good thing it worked out well!

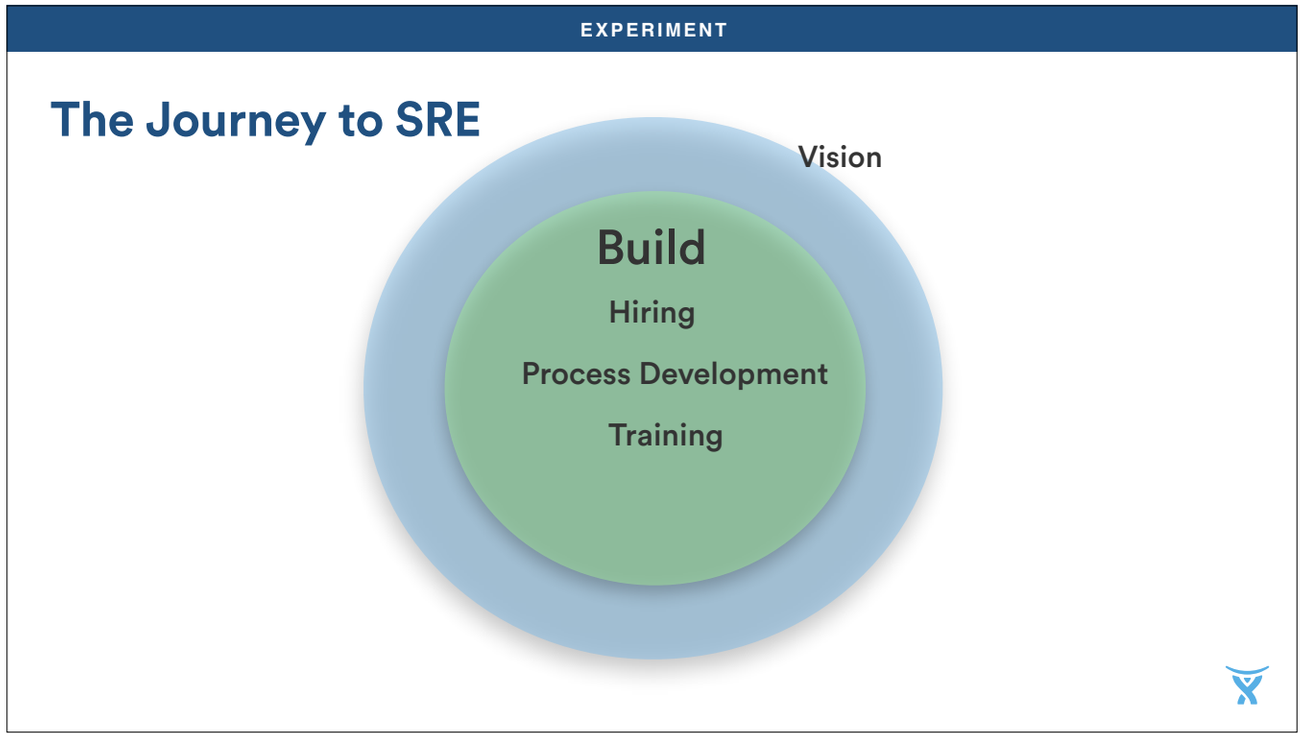
EXPERIMENT

The Journey to SRE

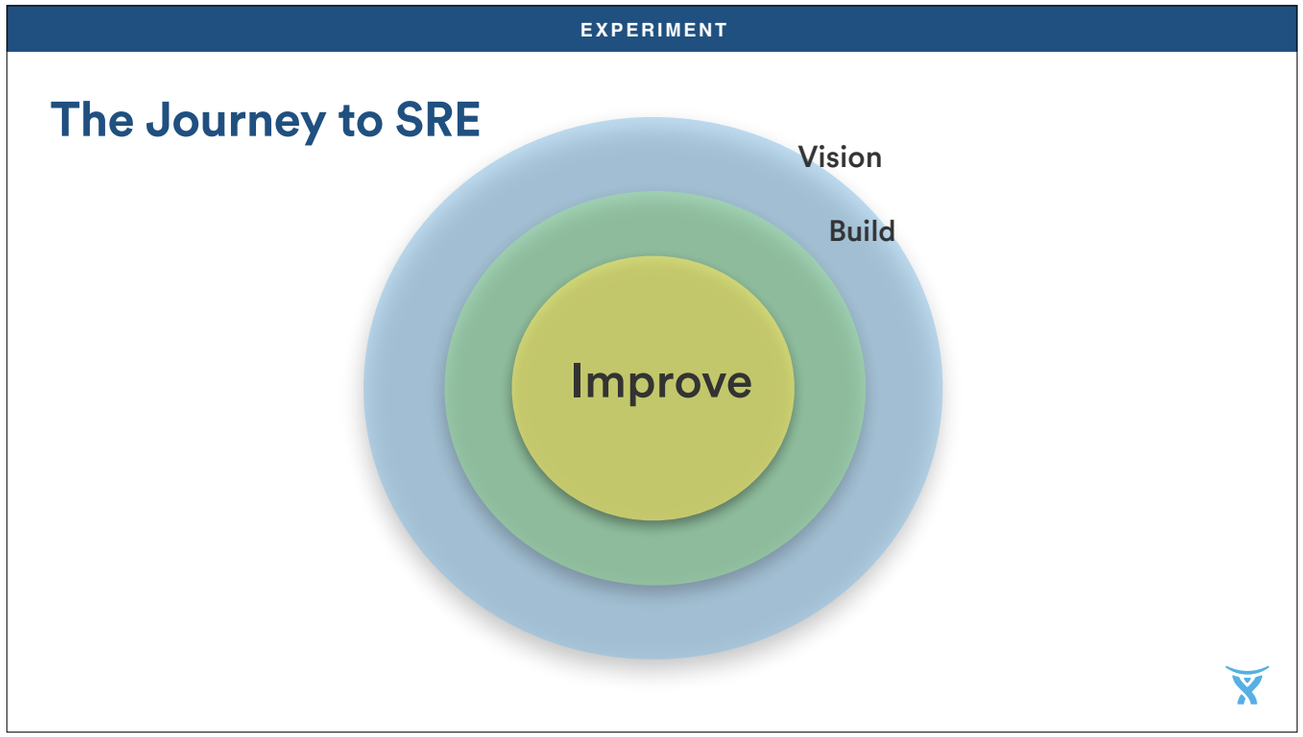
- Decided - Purchasing SRE
- Now we needed a team - not easy
- We spoke to people in the industry - no clear roadmap. Almost every organisation has to develop this for themselves around your culture.
- We made one up
- Upfront - we didn't have this when we started, we figured things out, then retrospectively put this together - we're using it for new SRE Teams
- **(*)**
- Vision - in detail.
- Spent some time on goals. Examples, Risk Register & Monitoring.
- Metrics - success criteria, trending.
- **(*)**
- Hiring, Training, logistics
- **(*)**
- Healthchecks each month for 6 months



- Decided - Purchasing SRE
- Now we needed a team - not easy
- We spoke to people in the industry - no clear roadmap. Almost every organisation has to develop this for themselves around your culture.
- We made one up
- Upfront - we didn't have this when we started, we figured things out, then retrospectively put this together - we're using it for new SRE Teams
- **(*)**
- Vision - in detail.
- Spent some time on goals. Examples, Risk Register & Monitoring.
- Metrics - success criteria, trending.
- **(*)**
- Hiring, Training, logistics
- **(*)**
- Healthchecks each month for 6 months



- Decided - Purchasing SRE
- Now we needed a team - not easy
- We spoke to people in the industry - no clear roadmap. Almost every organisation has to develop this for themselves around your culture.
- We made one up
- Upfront - we didn't have this when we started, we figured things out, then retrospectively put this together - we're using it for new SRE Teams
- **(*)**
- Vision - in detail.
- Spent some time on goals. Examples, Risk Register & Monitoring.
- Metrics - success criteria, trending.
- **(*)**
- Hiring, Training, logistics
- **(*)**
- Healthchecks each month for 6 months



- Decided - Purchasing SRE
- Now we needed a team - not easy
- We spoke to people in the industry - no clear roadmap. Almost every organisation has to develop this for themselves around your culture.
- We made one up
- Upfront - we didn't have this when we started, we figured things out, then retrospectively put this together - we're using it for new SRE Teams
- **(*)**
- Vision - in detail.
- Spent some time on goals. Examples, Risk Register & Monitoring.
- Metrics - success criteria, trending.
- **(*)**
- Hiring, Training, logistics
- **(*)**
- Healthchecks each month for 6 months

SITE RELIABILITY ENGINEERING

So how is this different from DevOps?

DevOps as a philosophy is much more sensitive to context than
SRE philosophy, because it grew from a broader collaborative base.

Charity Majors (@mipsytipsy) May 01, 2016 https://twitter.com/mipsytipsy/status/726756998201663489

To be clear: both approaches work for different problems.

- So at this point you're probably wondering well, what's the difference between DevOps and SRE.
- I want to make one point very clear: DevOps is a cultural solution and if you are implementing it to hit a reliability goal you may struggle.
- If you've got a problem with Reliability and learning from outages then SRE is for you. SRE is a focus on that particular problem, and has tools, methods and solutions for solving problems.
- The approach taken by the orgs I talked about before has had a lot of management support and buy-in for it to be successful.
- This isn't my unique thought - Charity Majors (a good ops-centric follow on Twitter) talks about it often and I encourage you to read her work on this topic.
- On that point though - let's delve into how we're tracking and what we've learnt.

So, how are we travelling?

- Almost 18 months on – some results and learnings i’d love to share

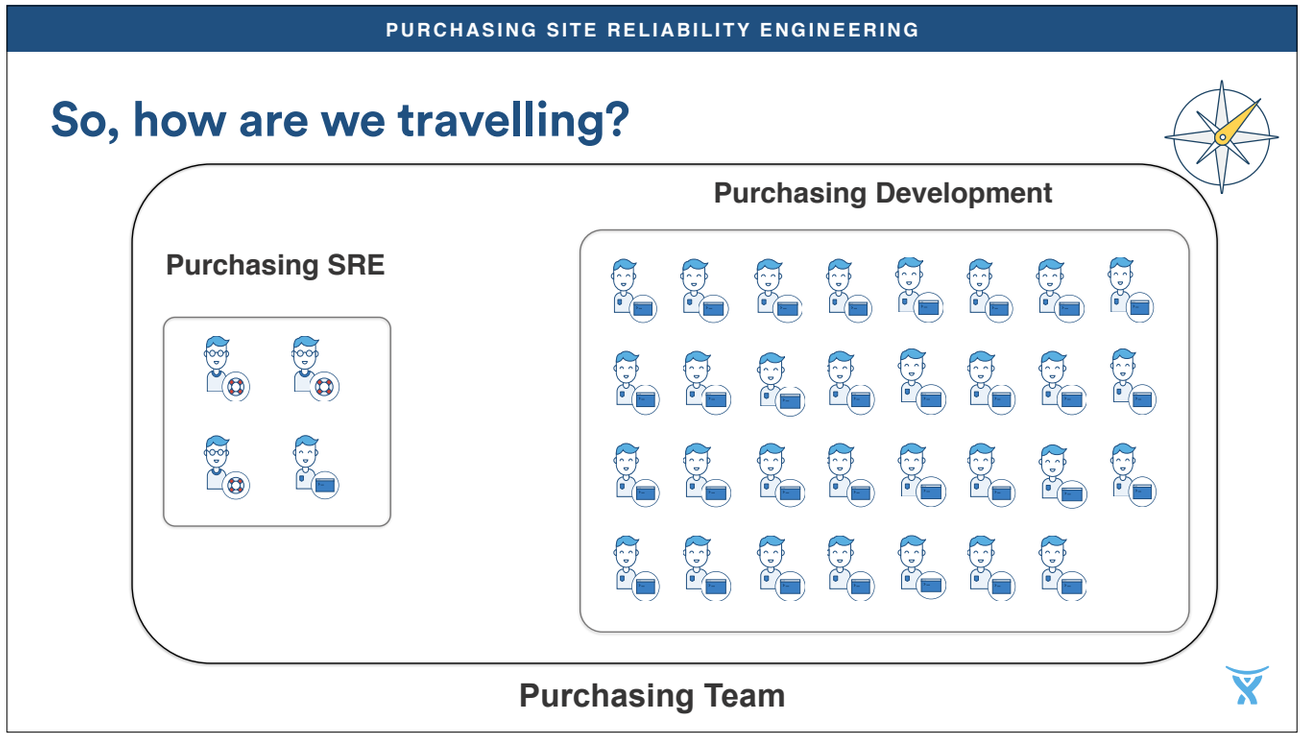
Rollout across key services

- Every key service that Atlassian runs now has aligned SRE staffing.
- Having a network or a guild of staff that share common problems has been instrumental in our ability to swarm on a problem.
- Case in point - we're not reinventing solutions when we have a problem. You have a method or a tool for speeding up incident resolution? We now can get that adopted by other teams with a decent amount of urgency.

- Every key service that Atlassian runs now has aligned SRE staffing.
- Having a network or a guild of staff that share common problems has been instrumental in our ability to swarm on a problem.
- Case in point - we're not reinventing solutions when we have a problem. You have a method or a tool for speeding up incident resolution? We now can get that adopted by other teams with a decent amount of urgency.

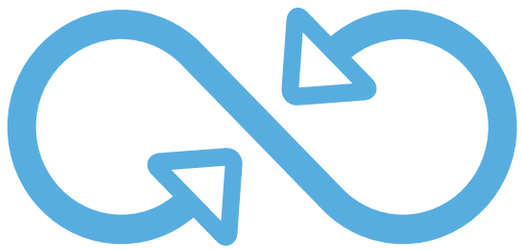
So, how are we travelling?

- Relationship with Dev Team
- SOPS and Purchasing Dev didn’t talk, only during incidents
- By design
- **(*)**
- Now, physically co-located
- Purchasing Dev on rotation - how we get them to do Ops work
- **(*)**
- How does this prove we have better alignment?
- We asked and i’d like to share responses
- **(*)**
- Alignment great, other accomplishments



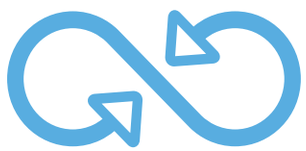
- Relationship with Dev Team
- SOPS and Purchasing Dev didn’t talk, only during incidents
- By design
- **(*)**
- Now, physically co-located
- Purchasing Dev on rotation - how we get them to do Ops work
- **(*)**
- How does this prove we have better alignment?
- We asked and i’d like to share responses
- **(*)**
- Alignment great, other accomplishments

So, how are we travelling?



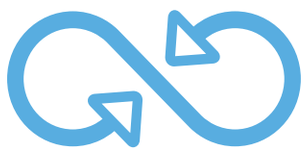
- Relationship with Dev Team
- SOPS and Purchasing Dev didn’t talk, only during incidents
- By design
- **(*)**
- Now, physically co-located
- Purchasing Dev on rotation - how we get them to do Ops work
- **(*)**
- How does this prove we have better alignment?
- We asked and i’d like to share responses
- **(*)**
- Alignment great, other accomplishments

So, how are we travelling?



- Relationship with Dev Team
- SOPS and Purchasing Dev didn’t talk, only during incidents
- By design
- **(*)**
- Now, physically co-located
- Purchasing Dev on rotation - how we get them to do Ops work
- **(*)**
- How does this prove we have better alignment?
- We asked and i’d like to share responses
- **(*)**
- Alignment great, other accomplishments

So, how are we travelling?

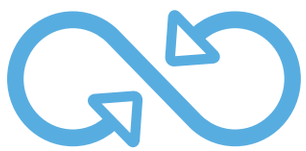


"The SRE team runs ahead of the rest of the team on reliability and **encourages everyone to lift their game**."

Andre S, Dev Manager

- Relationship with Dev Team
- SOPS and Purchasing Dev didn't talk, only during incidents
- By design
- **(*)**
- Now, physically co-located
- Purchasing Dev on rotation - how we get them to do Ops work
- **(*)**
- How does this prove we have better alignment?
- We asked and i'd like to share responses
- **(*)**
- Alignment great, other accomplishments

So, how are we travelling?



"The SRE team runs ahead of the rest of the team on reliability and **encourages everyone to lift their game**."

Andre S, Dev Manager

"In the past the separate ops and dev teams would often pick the solution they were best positioned to implement. I like that our SRE team is able to **pick the best solution to the problem instead**."

James B, Dev on Rotation

- Relationship with Dev Team
- SOPS and Purchasing Dev didn't talk, only during incidents
- By design
- **(*)**
- Now, physically co-located
- Purchasing Dev on rotation - how we get them to do Ops work
- **(*)**
- How does this prove we have better alignment?
- We asked and i'd like to share responses
- **(*)**
- Alignment great, other accomplishments

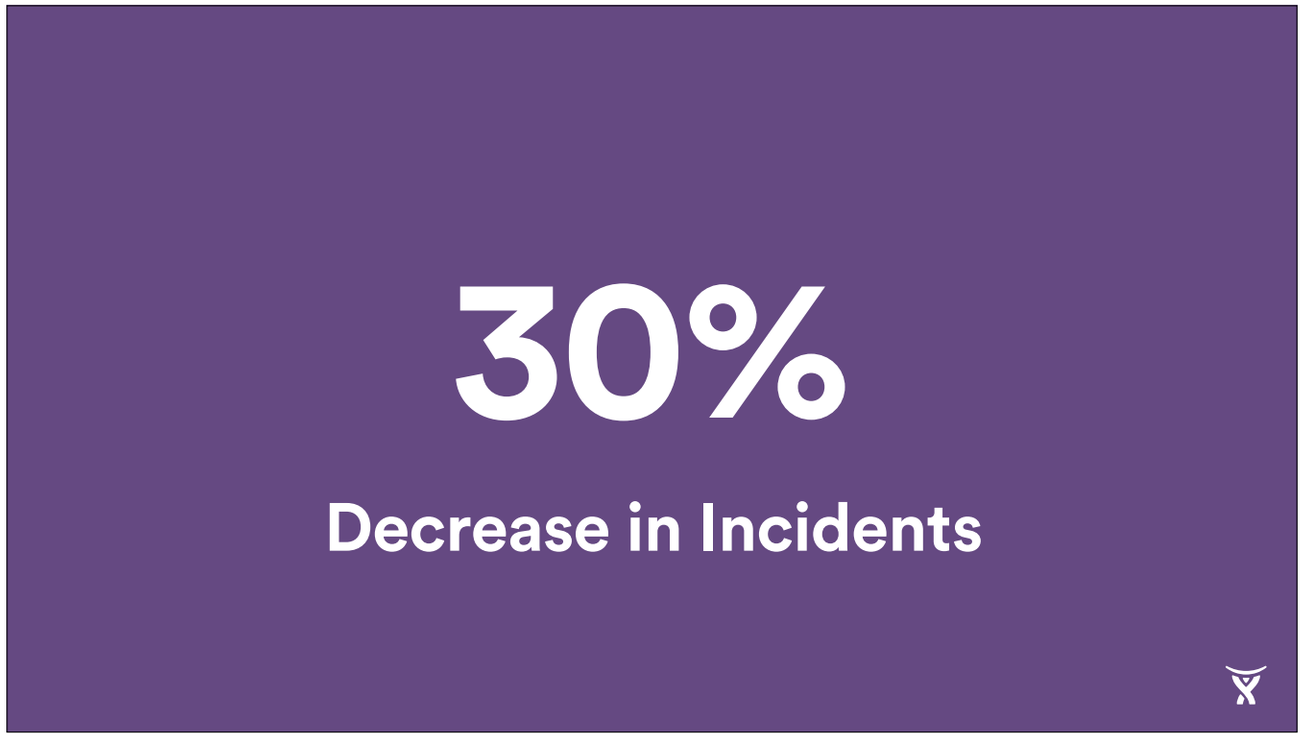
We are now completing 100% of our Post Incident Reviews for teams with To give you some perspective, we were probably completing about 10% before SRE, so this is a huge accomplishment for us. Follow up actions to prevent repeat incidents.

We tend to follow a pattern where we're preventing repeat incidents. This has been the main focus of our PIR process - we don't want to have the same incident occurring again with

To help with that, we have set up SLA's with buy-in from service owners to resolve the actions we come up with from the PIRs. For important actions (we call them Priority actions) we have a 4 weeks SLA for that action to be completed. We also have a lower tier of importance for actions that are more suggestions

More importantly, we've seen one of our key services (Bitbucket) go from having significant availability and reliability struggles on a daily basis to a smooth month in July, and largely continue on with that trend in August and September.

Reliability (that is the number of successful request responses from our services) have gone from 2 9's to 4 9's for our 5 key services. Like I talked about this in my previous slides, the focus on learning and evolving our service has been the critical learning that I suggest for you to take to your teams. If you aren't learning and changing what you do , then obviously you're going to keep having the same incidents.



As a result, and as a result of more advanced monitoring and alerting, we've reduced the number of incidents related to Purchasing Services by 30%.

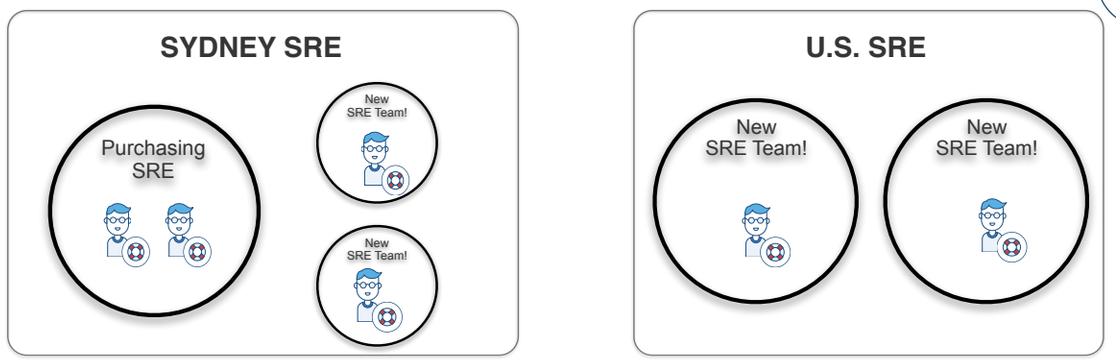
Next Steps

- What’s next?
- Success of Purchasing SRE, more teams!
- **(*)**
- Sops -> SRE, ops perspective
- Going to market - we’re hiring!

Next Steps

- What's next?
- Success of Purchasing SRE, more teams!
- **(*)**
- Sops -> SRE, ops perspective
- Going to market - we're hiring!

Next Steps



- What's next?
- Success of Purchasing SRE, more teams!
- **(*)**
- Sops -> SRE, ops perspective
- Going to market - we're hiring!

Summary

- Quick Recap
- **(*) – all**
- To be fair, there's a good chance we could have implemented different strategies to reduce incidents, automate things and improve reliability
- I'm sure many of you in the room have done within your own organisations.
- We strongly believe that Site Reliability Engineering contains the right formula for **us** to manage our services and deliver reliability to our customers.
- Thanks very much excited to share story and excited to answer any questions

IN SUMMARY

Summary

What is SRE?

- Solve problems with code
- Share Ops work with Devs
- Support Continuous Delivery - Error Budgets
- Post Incident Reviews
- DevOps principles
- Ultimately, align Ops and Dev

- Quick Recap
- **(*) – all**
- To be fair, there's a good chance we could have implemented different strategies to reduce incidents, automate things and improve reliability
- I'm sure many of you in the room have done within your own organisations.
- We strongly believe that Site Reliability Engineering contains the right formula for **us** to manage our services and deliver reliability to our customers.
- Thanks very much excited to share story and excited to answer any questions

IN SUMMARY

Summary

What is SRE?

- Solve problems with code
- Share Ops work with Devs
- Support Continuous Delivery - Error Budgets
- Post Incident Reviews
- DevOps principles
- Ultimately, align Ops and Dev

Where do I Start?

- Team Structure and Model
- Goals - build a backlog
- Metrics - measure success
- Hire coders
- Regular Healthchecks!

- Quick Recap
- **(*) – all**
- To be fair, there's a good chance we could have implemented different strategies to reduce incidents, automate things and improve reliability
- I'm sure many of you in the room have done within your own organisations.
- We strongly believe that Site Reliability Engineering contains the right formula for **us** to manage our services and deliver reliability to our customers.
- Thanks very much excited to share story and excited to answer any questions

IN SUMMARY

Summary

What is SRE?

- Solve problems with code
- Share Ops work with Devs
- Support Continuous Delivery - Error Budgets
- Post Incident Reviews
- DevOps principles
- Ultimately, align Ops and Dev

Where do I Start?

- Team Structure and Model
- Goals - build a backlog
- Metrics - measure success
- Hire coders
- Regular Healthchecks!

Our Wins

- Dev + Ops Alignment
- 100% PIR Completion
- DR Compliance
- Post Release Verification
- Incident Reduction

- Quick Recap
- **(*) – all**
- To be fair, there's a good chance we could have implemented different strategies to reduce incidents, automate things and improve reliability
- I'm sure many of you in the room have done within your own organisations.
- We strongly believe that Site Reliability Engineering contains the right formula for **us** to manage our services and deliver reliability to our customers.
- Thanks very much excited to share story and excited to answer any questions


Thank you for attending this session.
Please don't forget to complete an evaluation for this session!
Evaluation forms can be completed electronically on the FUSION 16 Conference App.
PRODUCED BY: itSMF USA HDI #SMFUSION