



SMARTPHONE &
TABLET GAMES
SUMMIT

DevOps for Mobile: It's Not Just For Enterprise Anymore

Tim Hargreaves - @timh
Release Engineer – Electronic Arts



GAME DEVELOPERS CONFERENCE March 14–18, 2016 · Expo: March 16–18, 2016 #GDC16





Introduction

Currently – Release Engineer @ EA Vancouver





FIFA®

For the Game. For the World.



Introduction

Previously – Build Engineer @ Roadhouse Interactive







Talking Points

- Why DevOps?
- What is DevOps?
- Comparison of Enterprise vs Mobile DevOps
- Mobile Specific Challenges
- Strategies



Why DevOps?

Goal

- building, testing, and releasing
- rapidly, frequently, and reliably



Why DevOps?

- Increase efficiencies
- Reduce time and effort spent
- Save money



What is DevOps?

- Intersection of existing activities
- Intersection of three traditionally separate departments



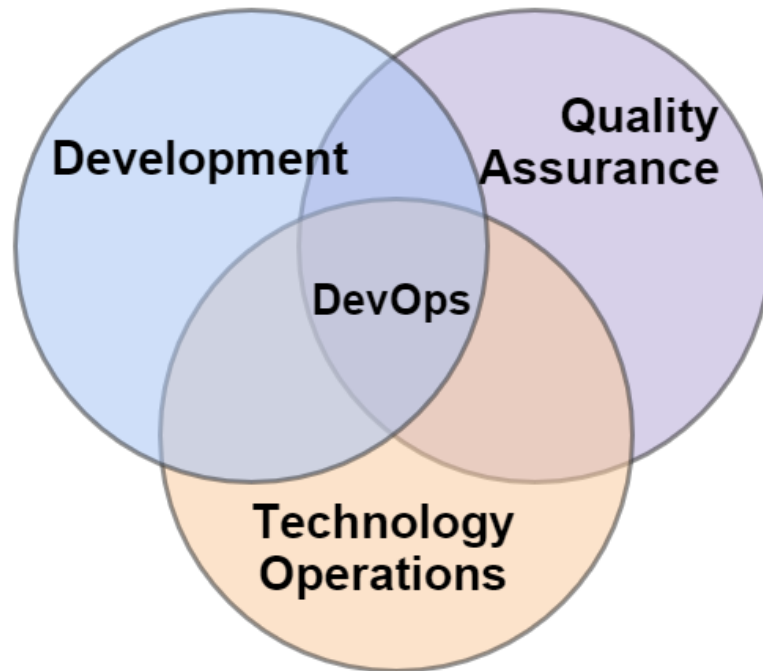
Intersection of Activities

- Development
- Quality Assurance
- Operations





Intersection of Departments





Who is DevOps?

- Your existing staff
- Cross trained with shared knowledge
- Still SMEs in their own field
- Can also have DevOps SMEs



What *isn't* DevOps?

- Band-Aid or bolt on
- A piece of software
- Replacement for Agile



What *is* DevOps?

- It aims at establishing a **culture** and **environment** where building, testing, and releasing software, can happen rapidly, frequently, and more reliably.

~Wikipedia



What is DevOps?

- **culture** and **environment**

- Need buy in from the frontlines

- Need support and encouragement from leadership



Put another way...

“DevOps is a human problem.”

~Patrick Debois



DevOps in Enterprise

- Documentation focuses on Enterprise space
- Even tools focusing on Enterprise space



DevOps in Enterprise

- Fast, small builds
- Web services
- Automated tests
- Deploy often and automatically
- Four browser targets
- Deploy to controlled infrastructure
- Push deployment



Challenges for Mobile Games

- Large, asset heavy builds
- Fat client
- Difficult to automate input/touch tests
- Long lag time in deployments
- Fragmented targets
- Deploy to vendor store
- Pull deployment



Strategies for Success

Version *Everything*

Goal: Time Machine



Strategies for Success

Version *Everything*

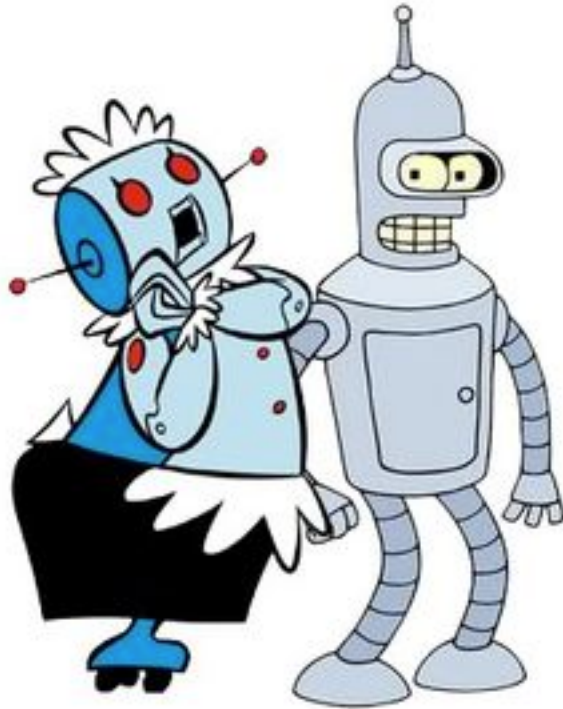
- Code and Assets
- Configurations
- Build Jobs
- Infrastructure setup/upkeep scripts

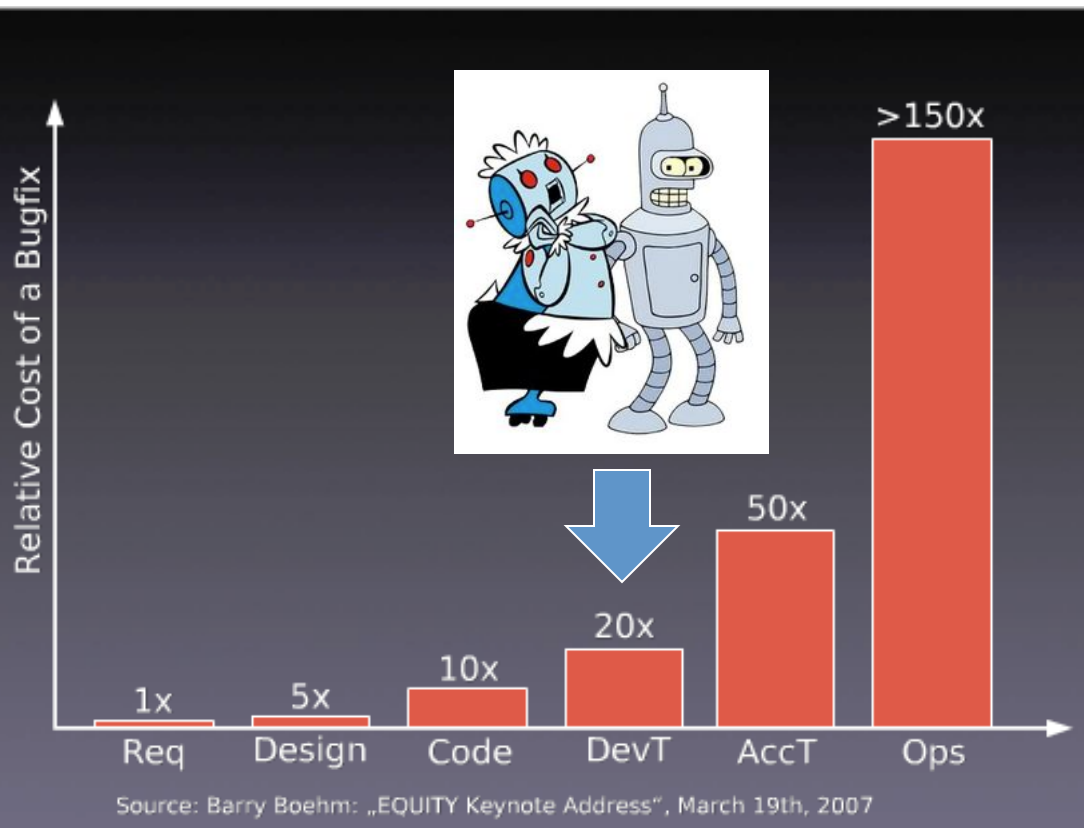


Strategies for Success

Automation

- “But we’ll only have to do this once*”
- *For extremely large values of once







Put Them Together

Version Control

+

Automation

=

Continuous Integration



Strategies For Success

Continuous Integration

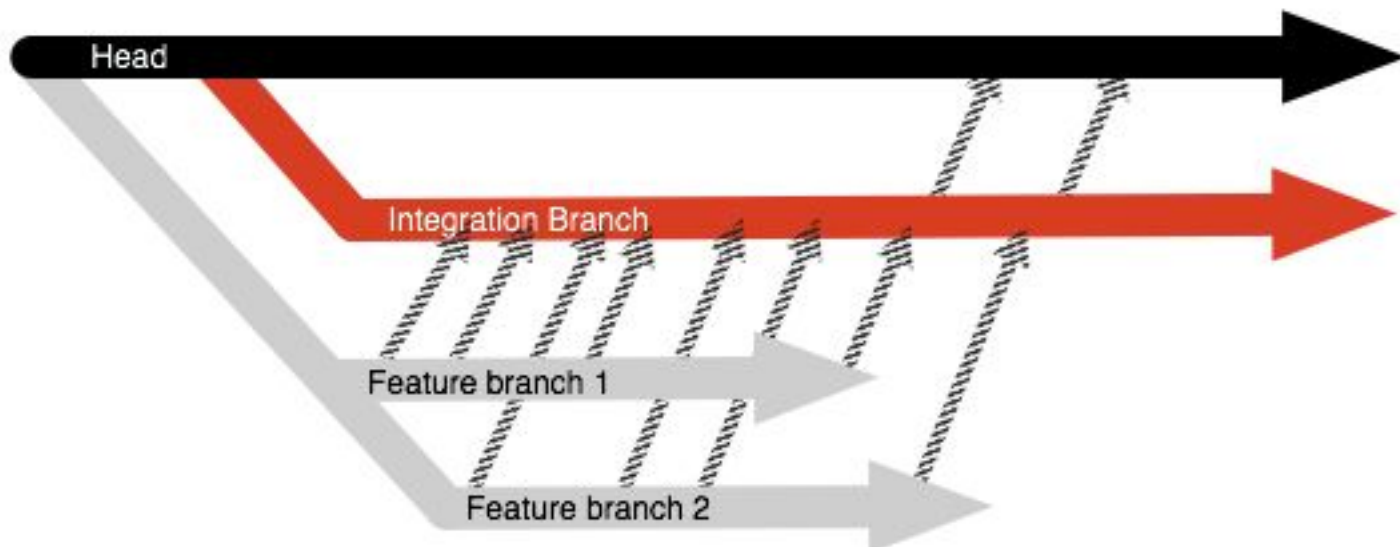
Beginner: Automatically poll SCM

Advanced: Automated branch integrations



Head

A large, solid black arrow pointing horizontally to the right. The word "Head" is written in white text inside the tail of the arrow.





Strategies For Success

Next Steps:

- Automated Testing
- Continuous Deployment



Challenges for Mobile Games – Revisited

Large, asset heavy builds

- Constant incremental builds
- Isolate changes (and build breaks)



Challenges for Mobile Games – Revisited

Fat Client

- Use local infrastructure
- Automated testing



Challenges for Mobile Games – Revisited

Difficult to automate input/touch tests

- Tools are coming along but...

...elbow grease for now



Challenges for Mobile Games – Revisited

Long lag time in deployments

- Prioritize Android
- Automated upload, no review process



Challenges for Mobile Games – Revisited

Fragmented Targets

- Automated testing (and builds)
- Device testing services



Challenges for Mobile Games – Revisited

Deploy to vendor store

- Well documented flow
- Task automation macros if you feel brave



Challenges for Mobile Games – Revisited

Pull Deployment

- Release a quality product often
- Alternatively, force server side upgrades



Bringing It All Together

We have

- An understanding of DevOps
- Techniques to overcome mobile challenges
- Ready and willing personnel ^{right?}



Challenges

- Inertia
- Comfort
- Control





Solutions

- Extreme DevOps
- Technical Enforcement
- Involvement



Final Thoughts

- **culture** and **environment** means changing **behaviour**
- Support with technology, training and teamwork
- If you just add a DevOps department, you've probably added another Operations department.



Contact

- Email
timhargreaves@jag.net
- Twitter
@timh