THE ENTERPRISE DEVOPS CHALLENGE

@royosherove

About Roy

- Elastic Leadership training for tech leads TOMORROW
- 20+ years in the software industry
- Most kinds of technical roles
- DevOps Consulting & Training to some of the worlds biggest companies

Agenda

- Influence Forces
- Enterprise Challenge(s)
  - Adoption
  - Metrics
  - Teams & Collaboration
  - Pipeline

THIS TALK WON’T MAKE A DIFFERENCE

*Unless we harness influence forces
INFLUENCE FORCES

But... Why?

If we understand:
- What influences us and our actions
- Our assumptions and how we got them

We can:
- Stop feeling stumped at the current situation (there’s always something to consider)
- Map challenges to Influence Forces
- Create a check list of actions and experiments
- Continuously experiment!

Influence Forces

Describe Behavior

- Find a symptom in the system (“releases take a long time”)
- “Releases take a long time”
- “There are many bugs in production”
- “It takes a long time to fix a bug”

- Discover root cause, and important behaviors you want to change

- Pick one core behavior that needs to change.

- 5 Whys
- Root cause -> core behaviors -> pick one
Describe Behavior

- Physical Behavior
- When we do X they do Y
- Choose a single behavior

- "When the team requests permission to change the pipeline, the IT admin says "nope."
- "When a release milestone arrives, the tech lead announces a code freeze."
- "When the build fails nobody on the team does anything about it."
- ...

Six Factors

- Personal Motivation
- Personal Ability
- Social Motivation
- Social Ability
- Structural Motivation
- Structural Ability

<table>
<thead>
<tr>
<th>Motivation</th>
<th>Ability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal</td>
<td>Does the person believe in the good behavior?</td>
</tr>
<tr>
<td></td>
<td>Is the person physically able to behave well, or physically avoid the bad behavior?</td>
</tr>
<tr>
<td>Group (Social)</td>
<td>Does the person follow others who spread bad behavior?</td>
</tr>
<tr>
<td></td>
<td>Do people react directly to the person about the bad behavior?</td>
</tr>
<tr>
<td>Company (Environment)</td>
<td>Does the company reward the bad behavior in some way?</td>
</tr>
<tr>
<td></td>
<td>Are the physical resources and objects around the person pushing for the bad behavior?</td>
</tr>
</tbody>
</table>
A checklist, not a recipe

ENTERPRISE CHALLENGES

Challenges

Inter-dependent → Monolithic

Approval Gates → Compliance

Waterfall → Manual, error prone processes

Challenges

Inter-dependent → Monolithic

Approval Gates → Compliance

Waterfall → Manual, error prone processes
Devops is change

- Existing workflows and assumptions
- Human behavior

Big Bang Transformation "Wishing"

Start

Finish
Big Bang Transformation “Reality”

People revert back to previous behaviors

Teach an Organization to Fix Itself
Enterprise DevOps Success Pattern

“Baby Step” Continuous Improvement
PILOT APPLICATIONS & TEAMS

Application Pilots

- Seed the beginning of a DevOps culture and teaming model through for (2) applications.
- Establish minimum viable set of automated tests.
- Establish a minimum viable DevOps tool chain
- Support the full end-to-end development lifecycle
- Establish a minimum viable velocity based policy for change management & compliance.
- A minimal set of DevOps metrics and baseline measurements.

ONE SIDED ADOPTION

Bi-Directional support

- Technical level support, training, coaching, leadership
- Middle & high level mgmt provide aerial support for changing behaviors
POLICY CHANGE

CD Pipelines are *(should be!)* automated policies of release

IDEAL DEVOPS WORKFLOW

Tiger Team Defines CD Change Management Standard

Existing Change Management Policy, Standards and SOPs

CM Tiger Team

Commercial DevOps Projects

Continuous Delivery (CD) Change Management Standard

Frequent, Compliant Application Releases

NEW
CHANGING WHAT MATTERS MOST

Value Streaming

END TO END VIEW (HOLISTIC)

Value Streaming

1. Identify actions taken in the value stream
2. Calendar time for each action
3. Calculate time actually worked in actions
4. Estimate time between actions
5. Identify any loop backs on average
6. Calculate Process Cycle Efficiency: Avg time Worked / Total Cycle Time

PCE = 509h / 3533h = 15.2%

7. Identify Waste
8. Identify and Prioritize Improvements
Example Value Stream (11-12 months) (efficiency: 15%, 50d)

<table>
<thead>
<tr>
<th>METRICS</th>
<th>RED</th>
<th>COVERAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local Metrics</td>
<td>RED = GOOD, COVERAGE = SUSPECT</td>
<td>Local Metrics</td>
</tr>
</tbody>
</table>
TIME FROM FIRST RED TO FIRST GREEN

DevOps Metrics
- Lead Time to Change
- Release Frequency
- Defect Rates
- Mean Time to Recovery
  - vs Mean Time to Failure

TEAMS & COLLABORATION

SHADOW DEVOPS
Shadow Conflicts

- Remote locations that need to coordinate (physical)
- "HQ" vs "Satellite" offices
- DEV lives in a bubble
- OPS hand-off
- Completely different tooling, processes...
- Re-run pipelines

WE DON’T HAVE TIME TO AUTOMATE

Vicious spiral

Less time to automate
manual work and test takes more time
more work added

Positive spiral

More time to automate
automated work saves time
more work added
Power is in the hands of the estimator

- Estimate work with quality built in (not optional)
- Don’t offer to remove quality/testing/etc.
- Stand by your estimates unless you are wrong, then change (as always)

CHOOSING TOOLS FIRST

Tooling Categories & Mapping

- Code mgmt. & Deploy
  - DVCS (bitbucket)
  - Code mgmt. & Deploy (Jenkins)
  - Test Automation (Jenkins)
  - Test Case mgmt.
  - Test/Release
  - Code Review (Bitbucket)
  - Artifacts mgmt. (Nexus)
  - Code Quality (SonarQube)
  - Secure Code analysis (Fortify)
  - Release mgmt. (Jenkins)
  - Config mgmt. (Ansible)

- Collaboration
  - Proj mgmt. (Jira)
  - Customer service mgmt.
  - CRM (Dynamics)
  - Wiki/Confluence
  - Doc mgmt. (Sharepoint)
  - Social Media (Yammer)
  - Chat (Skype)

- App Mgmt.
  - Server Monitoring (Nagios)
  - APM/Instrumentation (Dynatrace)
  - Patch mgmt. (BigFix)
  - Centralized App logging (Splunk)
  - Central Auth. (VAS)

- Infrastructure
  - IaaS (VMWare)
  - Public Cloud AWS, Azure
  - Private Cloud
  - Container mgmt. (Docker, kuber...)
  - Chargeback/showback (vReal)
  - VM Provision (terraform)
  - Service Discovery (consul)
  - Circuit Breaker
  - Reverse Proxy (F5 BigIP)
  - DNS (infoblox)
  - IP Mgmt (infoblox)
  - Server Inventory (Hubble)
  - Secret Management (Octopus)
  - SSL Cert mgmt.
Values

- Source controlled configuration
- Reliable/consistent
- Understandable/simple/friendly
- Compatible
- Automatable by other tools
- Exposes data to other tools
- Support/community
- Documentation
- Secure
- Speed
- Scalable/parallel work support

SUMMARY

Map challenges to Influence Forces
Create a check list of actions and experiments
Continuously experiment!

SHORT SONG?

THIS IS A TEST LINE
IT LOOKS LIKE YOU’RE DOING JUST FINE

IT’S TIME FOR A SONG OF MINE

SOME CONTEXT...

TRANSFORMATION SONG
It's a human sign

When transformations go wrong

It wasn’t supposed to end like this,
but,
now it's taking so long

No integration
No unit tests
Nobody's looking at the pipelines
and everybody's depressed
COLD COLD CLOUD
and empty queues
Nobody ever deletes the containers
Now Jira's locked out too

SACRIFICE
So we can finish in time
We just have to be ready
It's a victimless crime

So we must

Yes we must sacrifice
Quality would have been so nice
but I guess we must pay the price
that's all
Guitar playing.... (don’t hit space!)

Time to start a new project
Everything's looking great
This time we're gonna make it

so far we're just one day late!

Oh we can’t deploy to prod
Or any test VM
Did you send them an email?
What time is it in Vietnam?
COLD COLD SLACK
Empty chat room
Guess we have to wait so maybe
try tomorrow at noon

Guess we must
sacrifice
So we can finish in time
We just have to be ready
another victimless crime

Yes we must sacrifice
Quality would have been so nice
but i guess we must pay the price

Again.
THANK YOU

@royosherove
roy@Osherove.com
EnterpriseDevops.org (Training & Consulting)