

In-Flight Aircraft Mechanics

A Software Package Conversion Project

Philip J. Hollenback
philiph@yahoo-inc.com

November 2010

What We Started With

- 7500 production servers
- Nondeterministic software pushes
- Automated install of dependencies
 - same as any other packaging system

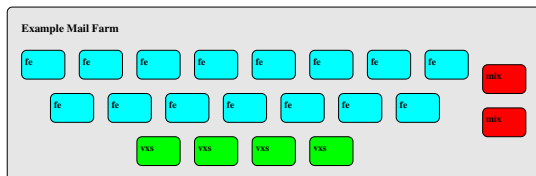
Goals

- Repeatable software pushes
- Exact specification of installs
- All Apps All Environments
 - Every machine gets every package!
- Rollback to previous releases

Tools

- Tools already existed at Yahoo
 - don't reinvent the wheel
- Package management: yinst
- Servers state definition: igor
- Release assembly: SRM
- Software deployment: Pogo

Architecture



- 'the farm' - about 20 servers
- mailstore and access
- where a user's mail resides
- mostly independent from other farms

Where to Begin?

- map packages to functional areas
 - dev team signoff
- find hidden package deps and assumptions
- do we really need 1300 packages?
 - sadly, yes

Modeling

- VM test environment
- install package list with igor
- did it work?
 - not the first 50 times!
- rinse, lather, repeat

Production Conversion

- rolling upgrade on live hosts
 - OS reinstall required!
- 6 person team (SNV/BLR)
 - 1 scout, audit ahead of others
- 9 months work total
- Lots of IRC & late nights
 - shared between US & Bangalore

Release Assembly

- dev groups submit packages/settings as igor tags
 - tags specify role revision
 - exact list of packages/settings
- RM team assembles into release using SRM
 - also performs basic testing
- QA does functional testing
- QA Signoff - we have a train!
- Repeat every 3 weeks forever

Production Deployment

- Release Management Team responsible for schedule
- currently about 10 days to completion
- 4 stages
 - 1 farm
 - 100 farms
 - 200 farms
 - 200 farms
- Stop and rollback at any point if needed
 - yes it does happen every few months!

What Worked Great

- Defining system state at the package level
- All Apps All Environments
- Rollback!

Speaking of Rollback

- Yes, it does work!
- Requirements
 - extensive up-front release testing
 - can't rollback in production without testing in QA!
 - commitment to quality packages
 - decoupling packages from settings
 - central config servers

Do-Overs

- Create audit tools first
- Create push tool first
 - pogo came along 6 months after start
- Reduce vulnerable mixed-state window during conversion
- Set up a longer release cycle
 - started with two weeks, now three

Ongoing Issues

- Continuing to remove configuration from packages
 - Yinst settings
 - Central config servers
- Igor sync monitoring
- Deal with emergencies more smoothly
 - Best case release push to all hosts: several days!

To Sum It Up

KEEP IT STUPID, SIMPLE!

- Install the same packages everywhere
- Don't use inheritance for system state
- Basically just don't be clever!

Thanks for Listening!

Phil Hollenback
philiph@yahoo-inc.com
@philiph