# DevOps at Sonatype

# Contents

# 1    DevOps at Sonatype

Lessons Learned from Using Repository Managers and Software Supply Chain Tools

Manfred Moser - @simpligility

• Author, Trainer and Community Advocate

Brian Fox - @brian_fox

• VP of Product Engineering

Sonatype - @sonatype - www.sonatype.com

# 2    What Are You Learning Today?

• See what worked for us

• And what caused us problems

→ Improvements for your own situation

# 3    What does Sonatype actually do?

Manages and develops:

• Nexus repository manager

• Nexus Lifecycle software supply chain automation

• Nexus IQ Data Services component intellignec

• Central Repository (formerly Maven Central)

• OSSRH and related Forges

• Related documentation, websites, blogs, videos,. . .

# 4    Nexus Repository Manager

• Nexus 2 - 70% market share, >50K server installations

• Nexus 3 - upcoming

• Open source: Maven2, npm, RubyGems, NuGet, Docker, . . .

• Commercial: Staging, Smart Proxy, . . .

## 5   Software Supply Chain Automation

- Nexus Lifecycle

- Nexus IQ Server

- Partner integrations (HP Foritfy On Demand ?)

- Nexus Firewall (?)

- Integrations with Jenkins, Hudson, Bamboo

- Eclipse IDE/M2e Plugin

- SonarQube Integration

All of this is backed by. . .

## 6   Nexus IQ Data Services

Curated and managed data and services for:

- Age

- Popularity

- Security vulnerabilities

- License analysis data (observed and declared) *

## 7   Central Repository

- Largest Maven2 format repository (numbers!)

- High performance, global CDN

- > 17 billion download in 2014

- Default in Maven2 and others

And the components come from

## 8   OSSRH and Forges

Input funnel for Central Repository

- OSSRH - large deployment of Nexus Pro

- Apache, JBoss, . . .  - secondary Nexus instances that sync

- Community support - onboarding and documentation

- TBD add some numbersf

# 9 Who Helps at Sonatype

- Internationally distributed (West coast to east coast to eastern urope)
- Multiple-time zones
- Remote work the rule, not exception

# 10 Teams

- Numerous smaller teams
- Different focus of teams
- Cross-team members

# 11 Process

- Scrum framework
- Backlog refinement

# 12 Communication

- Google Drive
- HipChat
- Good old phone and VOIP
- Google Hangouts
- join.me

# 13 SCM

- GitHub
- Atlassian Stash

# 14 Track and Plan

- JIRA
- Aha
- Trello

# 15 Continuous Integration

- Bamboo
- feature branch builds

# 16 Build

- Apache Maven
- Grunt and NPM for client side
- Bash Scripts

# 17 Maven Tips and Tricks

- Maven wrapper
- Follow
- Find balance of number of projects vs.
- Avoid forking third party libraries

# 18 Develop

- Feature branches
  - with CI
  - short lived
- IDE
  - Eclipse IDE
  - IntelliJ IDEA
- Lots of OSX, some Windows & Linux

# 19 Test

- Geb
- Spock

# 20 Document

- Atlassian Confluence
- Google Docs
- Asciidoc
- Pelican

## 21   Continuously Build

- Atlassian Bamboo

- > 100 build plans

- Elastics Bamboo - EC2 instances

- Feature branch builds increases number

- Automated functional test suite runs

- Automated release

- Documentation builds and deployments

## 22   Bamboo Tips

- Resource project (link to video maybe)

- Fresh Maven repo for each build off Nexus

- Build plan notifications into HipChat channels

- Links to GitHub and JIRA

## 23   More Bamboo Tips

- Share repo and other outputs as build artifacts

- Static documentation = usable artifact

- Limited number of standard Amazon Machine Images (AMI)

- Include standard tools

- Build artifacts stored on Amazon Elastic Block Storage (EBS)

- Base build plan with tool configuration\

## 24   Build Plan Commonalities

All builds plans:

- Common configuration from base plan - used as shared artifact, managed in git repo

- Global variables - defaults that allow overrides

- *build* task - compile and test code.

- *release* task - publish to Nexus and tag in git

- bundle test artifacts

- Main vs features branches - different config

- Branch builds auto-created

## 25   Co-Locate For Performance

Repository manager * component source for consumers * component target for producers

images/producers-consumers.png

CI does both, so

Best practice - Get it close

images/nexus-bamboo-rso.png

And sync to another repository if needed.

## 26   Validate

- SonarQube - integrated in Bamboo and GitHub

- License check with Maven plugin

- No merges without build passing and code review

- Nexus Lifecycle TBD

## 27   Release

- Workflow and notification with Nexus staging

- Including validation with Nexus Lifecycle

  - Security checks
  - License checks
  - ?? anything else, maybe component age

- Usage of release build number - `2.11.4-01`

- Same release stuff on OSSRH

## 28   Manage Components

Nexus Repository Manager and Nexus Lifecycle Nexus Repository Manager * SNAPSHOT versions of master are deployed to Nexus * feature branch versions are not deployed * master = integration branch * within build cluster * also used by developers * target for deployments * proxy * host * staging * Thousands of users and projects on OSSRH * smart proxy to Nexus outside cluster

## 29   Nexus Lifecycle

- Visualize risk through rule-based automation

- Streamlines component selection with real time data

## 30  Starting Off

- Define risks we care about (internal, external, customer used)

- Open source contributions change out policy (since we contribute it makes sense to use bleeding edge)

- Understand our process and tooling

- limit overhead in our build automation

## 31  Nexus IQ Server Deployment

images/nexus-iq-server-integration.png

## 32  Policy

images/sonatype-policy.png

## 33  Resulting Report

images/nexus-clm-report.png

## 34  Nexus Lifecycle Lessons

- Surprise how many components are used - so many!

- Blocking a release for policy violations

  - is a big stick
  - but it works

## 35  Deploy

Ops team:

- RPMs

- Docker images

## 36  Support

- Engineers on support team

- Write lots of automation and other code

- ZenDesk

## 37 Operations

Saas is used whenever possible Kanban process iDoneThis

Nexus used as binary blog store e.g. insert image with Ansible from JasonSwank

Our Nexus instances we manager vary from hundreds of GB to terabytes of non-proxied context

## 38 Operations - Service Management

Using Ansible and Nexus:

images/service-management.png

## 39 Community

- Actively work with vendors

- Including open source projects

- Help upstream to help yourself

    – Report issues
    – Release testing
    – Contributions

## 40 Next?

- Join the Nexus community

- Star using Nexus OSS

- Try Nexus Repository Manager

- Try Nexus Lifecycle

- Come to our booth

## 41 The End

Questions, Remarks & Discussion

---

**Tip**
Slides and examples at http://sonatype.github.io/nexus-presentations/ or email manfred@sonatype.com

---

i. and we are hiring!

## 42 Resources

- sonatype.com
- Nexus community
- Central Repository and documentation
- Youtube channel
- Inside Engineering - blog post
- Inside Engineering - videos
- Nexus Tips from the Trenches video series