Simian

Enterprise Mac OS X Software Deployment

John Randolph and Justin McWilliams
Software Engineers @ Google

What is Simian?

munki + Google App Engine

Munki Overview

- Developed and open-sourced by Greg Neagle

- First released in early 2009

- Used by diverse community of admins

- Deployments from tens to hundreds of Macs

- Munki was an obvious choice for Google

Munki Key Features

(for Enterprises)

- Deploy new packages

- Update third-party software

- Background (unattended) package installation

- Force-reboot after a defined time

- Apple Software Update integration

Munki Server Options

No provided server. Munki is designed to work with any web server, e.g. Apache

Server only hosts static control files for clients

Client makes standard HTTP requests

Munki Static Configs

Munki config management is manual

Admins use CLI tools to generate static control files and place on server

Administration Overhead

Apache

Apache security updates

hostname based configs

custom catalogs

custom manifests

shell scripts

CLI file generation

editing on server

This works well
for smaller sites,
but ...

scale?

storage replication

storage backups

load balancing

XML editing

flat files

clients steered to configs
manually

re-group clients into new pkg
groups from remote?

Simian

Web service that dynamically generates and serves control data

No stored files

Google App Engine Hosting

- No setup or management of physical infrastructure
  - Web servers, database servers, replication, etc.
  - Distributed, load balanced, backed up, etc.

- Economic hosting model
  - Only pay for what you use

App Engine Scale

- Scale with growing enterprise
  - More Macs added over time? GAE scales up automatically
  - Load tested 75k+ virtual clients
  - Google Example
    - Google has tens of thousands of Macs
    - Overnight growth

App Engine Accessibility

- Internet Facing

- Securely manage clients outside internal/corporate network or VPN

  - Laptops

Simian

- Open-sourced in January at Macworld 2011

- Basic reporting

  - Client versions, connection logs, install status, pending installs, etc.

- Dynamic request handling == flexibility

- Last code release in March

New open-source release available TODAY!
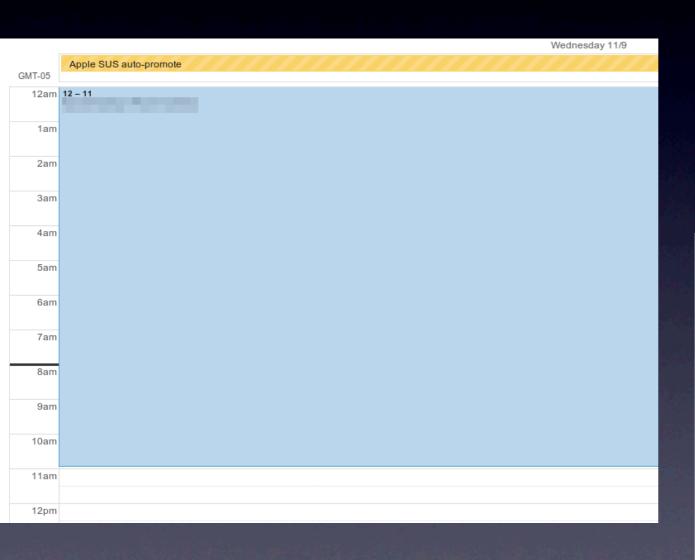
Apple Software Update Catalog Management

- Replace your Apple Software Update infrastructure

    - Kill your XServe(s), we killed ours!

- Daily catalog sync

- Admin notification of new Apple updates

- Optional

Apple Software Update Catalog Management

Name	Version	Product ID	Apple Release	Testing Auto Promote Date	Stable Auto Promote Date	Manual Override	unstable	testing	stable
Thunderbolt Software Update	1.1	041-3123	Oct. 28, 2011, 1:54 a.m.		None	☑	☑	☑	☐
Aperture Update	3.2.1	041-2856	Oct. 27, 2011, 7:59 p.m.		Nov. 9, 2011	☐	☑	☑	☐
EPSON Printer Software Update	2.9	041-2305	Oct. 27, 2011, 7:55 p.m.	Nov. 2, 2011	Nov. 9, 2011	☐	☑	☐	☐
HP Printer Software Update	2.8	041-2167	Oct. 27, 2011, 7:55 p.m.	Nov. 2, 2011	Nov. 9, 2011	☐	☑	☐	☐
MacBook Pro EFI Firmware Update	2.3	041-2515	Oct. 26, 2011, 8:18 p.m.	Nov. 2, 2011	Nov. 9, 2011	☐	☑	☐	☐
Mac mini EFI Firmware Update	1.4	041-2517	Oct. 26, 2011, 8:18 p.m.	Nov. 2, 2011	Nov. 9, 2011	☐	☑	☐	☐

- Apple updates auto-promote, admin override
- ETA Q1: background/unattended, force-reboot

Apple Software Update iCal Feed

Dynamically Target Clients

Munki manifests are targeted lists of available packages

Dynamically add or remove packages based on:

- host uuid
- owner username
- site name
- os version
- future expansion...

Dynamic Manifest Mod API

Google Example: Internal commercial software request/approval system

1. Employee requests software in ticketing system

2. Manager approves

3. Software is made available to the employee without any tech interaction

Broken Clients

- Request logs be uploaded during next preflight
  - Notified via email when logs are uploaded

- Clients self-heal, early on in preflight
  - Server detects need based on preflight/postflight date comparison
  - Server instructs the client to download and install the latest version

Additional Reporting

- **Average Install Duration + Install Count**

- **Managed Software Update GUI action reporting**

- Apple Software Updates reported separately

- Manifest Modifications; view Dynamic Manifests

- Install Reports cache while offline

- Lost/Stolen client connection IP logging

- UserSettings Knobs

Demo

Source Code / Documentation / etc
http://code.google.com/p/simian
patches welcome

Email Lists
simian-{announce,discuss,eng}@googlegroups.com

Google+ Hangout Office Hours
Alternating Monday 11am-1pm Eastern

Come see us in the Scripting Lab