OUR SYSTEMS
BACKGROUND

• 2001 ~ 2005: 3 shared servers
• 2005 ~ 2009: n VPS
• 2009 ~ today: m physical servers (20 < m < 30)
• today ~: \(\infty\) cloud servers
BUT...

This was meant to be a nightmare
Automate, ok...

but WHAT to automate?
APPLICATION DEPLOYMENT

zc.buildout, fabric, ...

✔ Already achieved
OUR NEEDS

1. Server provisioning
2. Configuration update/change
ANSIBLE TO THE RESCUE
Ansible is a radically simple IT automation engine that automates cloud provisioning, configuration management, application deployment, intra-service orchestration, and many other IT needs.
ANSIBLE

- Write a YAML file
- Run it
- Everything done
[cheers and applause]
- hosts: all
tasks:
  - name: Add user setlem
    user: name=setlem state=present shell=/bin/bash

  - name: Ensure additional packages via apt
    apt: pkg=nginx state=present
EXTENDING ANSIBLE: ANSIBLE GALAXY

- pypi for ansible
- Roles for almost anything: package configuration, application deployment, swap files, exim configuration, ...
OUR ANSIBLE SETUP

We ended having 5 main configuration files for our servers:

- Common setup
- Setup for Plone project
- Setup for Django projects (Common + Mysql)
- Setup for WordPress projects (Common – nginx + Apache + PHP + Mysql)
- apt package update (we use Debian)
OUR FIRST TIME

- After some tests, we run it on production
- 4 servers with special firewall rules
- 10 servers with old ssh keys
- n incorrectly configured exim mail servers
DYNAMIC INVENTORIES

- Dynamic list of hosts
- No need to keep a stone-written list of hosts
- Everything automated
CURRENT STATUS

- We create all our servers using Ansible
- We regularly run Ansible to ensure that nobody has changed the configuration
- We keep updating our Ansible config file to address a few glitches
(Small) Problems

- Server specific setups
- Different firewall rules
- Domain-dependant mailserver configuration
- Project specific requirements
FUTURE DEVELOPMENTS

- Automate Plone hotfix installing
- Automate django deployments
- Automate nginx configuration
- Autocontained projects: server setup + application deployment (DevOps culture)