Monolithic installs VS single responsibilities networks

Fedir RYKHTIK

DrupalDeveloperDays 2015

#drupaldevdays

About speaker

Fedir RYKHTIK

CTO @AgenceStratis (Toulon, France)

Drupal developer since 2008

DrupalMeetup organizer

White elephants handler

@FedirFr

Talk summary

The problem

Sometimes drops begins to be too big

Big heavy sites

- Some times not stable
- Lots of hooks
- Lot’s of operations
- Lot’s of usage scenarios
- Difficult to update
- Difficult to migrate
- Difficult to scale

99% of time You don’t need ALL modules / hooks at the same time

Drupal’s Site Usage Scenarios

- Content administration
  - Permissions
  - Workflow
    - Validation
    - Content programmation
- Content automatic importing
- Exporting content (REST/RSS)
- Showing the content

Microservice concept

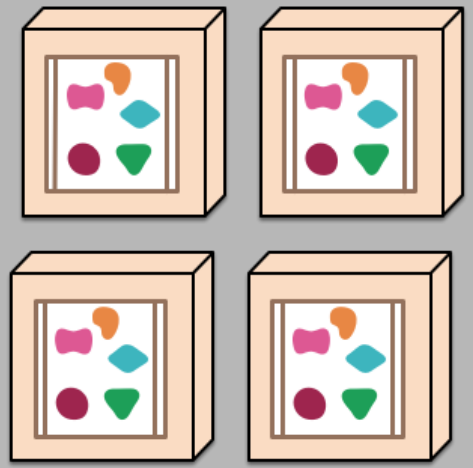
Microservice

In computing, microservices is a software architecture style, in which complex applications are composed of **small, independent processes communicating with each other using language-agnostic APIs**. These services are **small, highly decoupled** and **focus on doing a small task**.

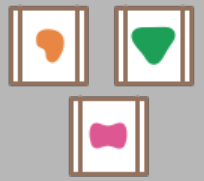
http://en.wikipedia.org/wiki/Microservices

A monolithic application puts all its functionality into a single process...

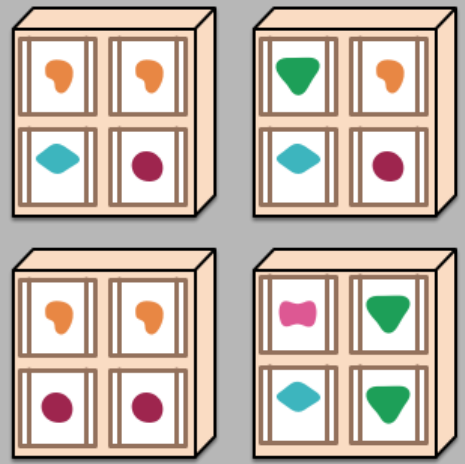
... and scales by replicating the monolith on multiple servers



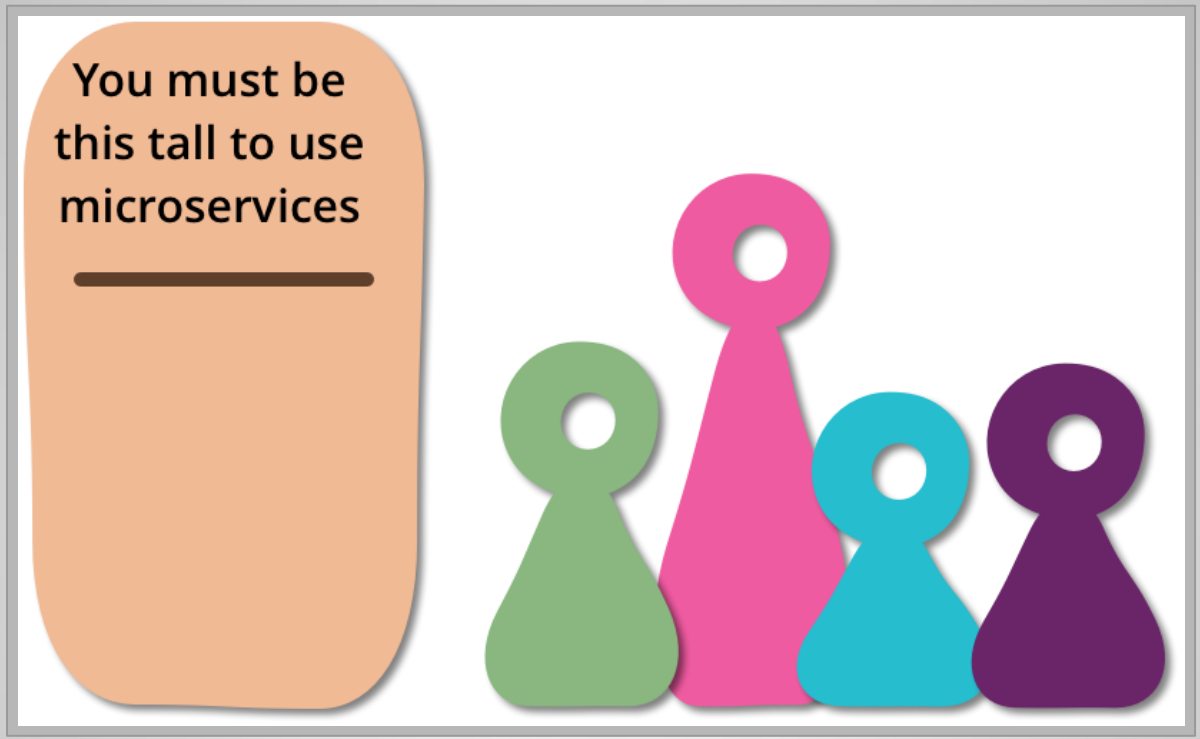
A microservices architecture puts each element of functionality into a separate service...



... and scales by distributing these services across servers, replicating as needed.



http://martinfowler.com/articles/microservices.html



*“There are certain things you need to get sorted out before you can put your first microservices system into production: **monitoring**, **provisioning**, and a **devops culture**.”*

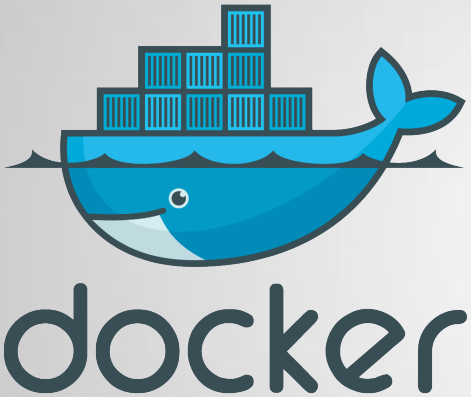
Martin Fowler

http://martinfowler.com/bliki/MicroservicePrerequisites.html

Control the swarm

Provisioning

DevOps tools



ANSIBLE

ANSIBLE

TOWER

System Architecture description

- Vagrant wrapper on local machines
- Docker + Ansible in production
- Ansible Tower for efficient control

Monitoring

Efficient logging for the best visibility & collision detection

- LogStash
- Kibana
- monolog
- Drupal’s “Log HTTP” project

HowTo

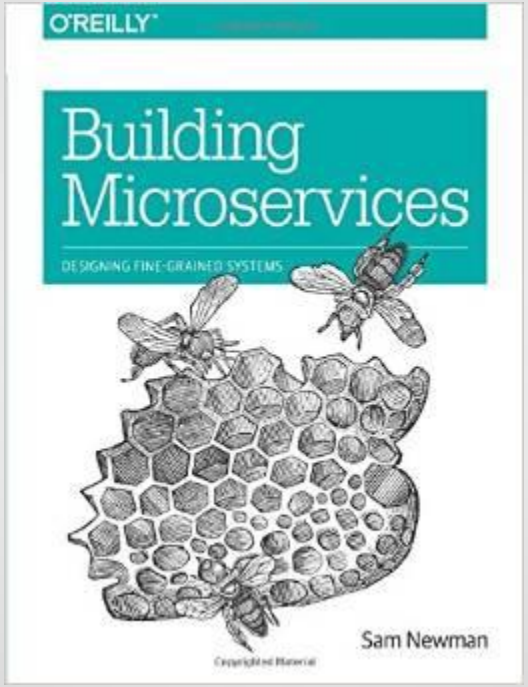
Define service roles

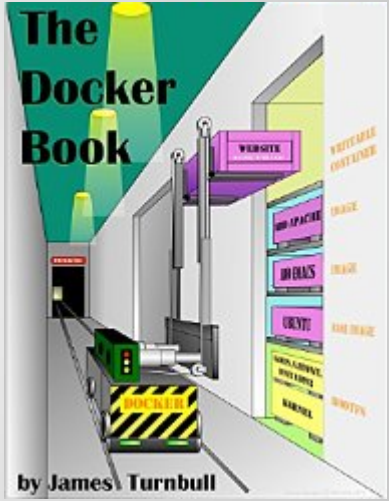
- Content import
- Content administration
- Content rendering
- Front end caching
- User management

Transformation of an existing site into microservice network

- Find a white elephant
- Detect the services in use
- Describe flows of connection
- Create the network via DevOps tools
- Create & test standalone services
- Connect flows
- Pack it up
- Deploy

Books to read





Links

- http://en.wikipedia.org/wiki/Microservices
- http://en.m.wikipedia.org/wiki/Service-oriented_architecture
- http://fr.slideshare.net/TylerTreat/from-mainframe-to-microservice-an-introduction-to-distributed-systems-41004778
- http://martinfowler.com/articles/microservice-testing/
- http://martinfowler.com/bliki/MicroservicePrerequisites.html
- http://martinfowler.com/articles/microservices.html
- https://www.loggly.com/blog/logs-for-drupal-why-you-need-them-and-how-to-do-it/
- http://www.slideshare.net/slideshow/embed_code/key/464TyWARPmjno8
- https://wooster.checkmy.ws/2014/04/elk-elasticsearch-logstash-kibana/
- http://odolbeau.fr/blog/when-monolog-meet-elk.html
- http://fr.slideshare.net/jamtur01/yes-logging-can-be-awesome
- https://github.com/victorlin/ansible-docker-demo

Q&A

Any questions ? Let’s talk ! :)

Thank You