

An Evolutionary Hierarchy for Controlling Large-Scale Distributed Systems

*Damien Anderson, Paul Harvey, Yusaku Kaneta, Petros
Papadopoulos, Philip Rodgers, Marc Roper*

Presentation Overview

Case Study

Content Delivery Networks

Implementation

Initial Results

Conclusions and Future Work

Testing Ideas in the Real World

Case Study

Case Study

Does the proposed Architecture work?

How can general problems be encoded for a genetic algorithm?

How can a general fitness function be defined?

How do you start to develop an autonomous system for any type of network or large scale system?

The Answer

Simplify the problem

- Develop an autonomous system for one type of network
- Learn lessons from this to apply to the wider case

This allows the exploration of the problem in a concrete, real world example

You got content and need to deliver it? You need...

Content Delivery Networks

Content Delivery Networks (CDN)

- Deal with user requests for content around the world
- Responsible for delivering high quality user experience for online content
- An Efficient CDN reduces load on central systems and delivers content more smoothly to users

Varnish Cache

A cache solution which can be used to build a CDN

Real world application

Dynamic configuration and high level of customisation



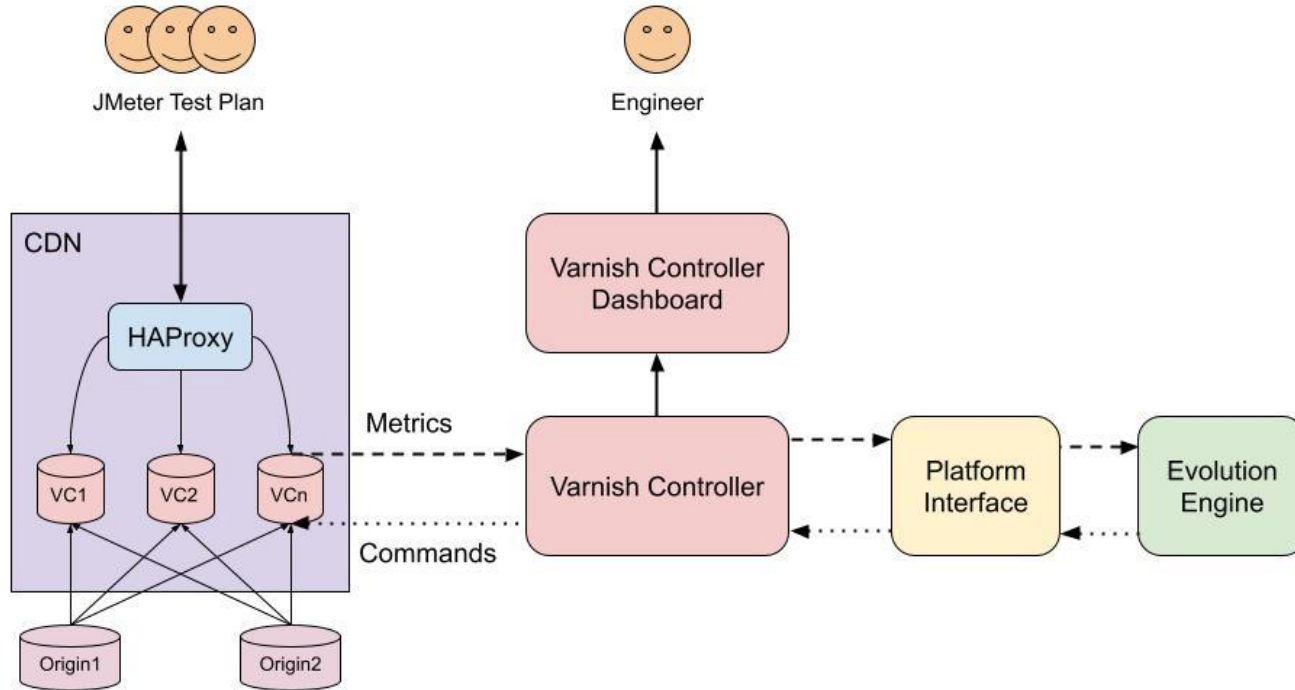
Varnish Configuration Language (VCL)

- Domain Specific Language for configuring Varnish Cache
- Huge amount of customisation options
- VCL files store configuration settings for Varnish Caches
- Can be configured dynamically, and over HTTP

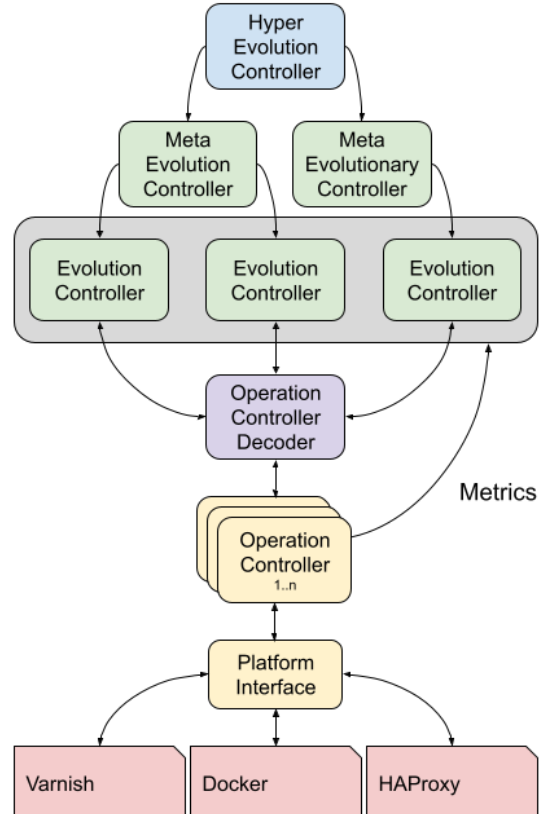
Can we build it? Hopefully

Implementation

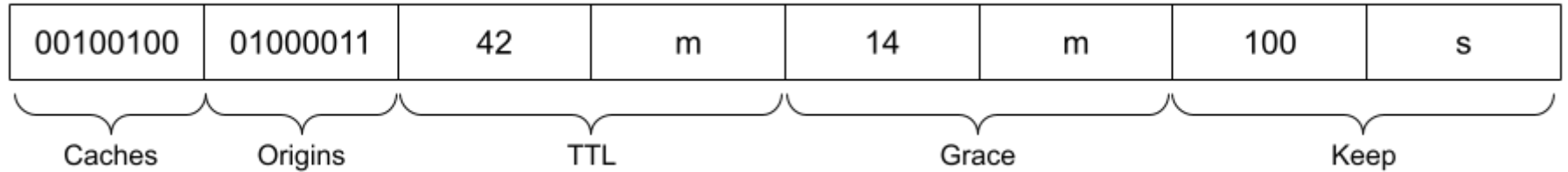
Experimental Setup



Evolutionary CDN System Architecture



Problem Encoding



Fitness Function

$$\text{Util for a single cache node} = \frac{\text{SLA hit rate for content}}{\text{Average hit rate for all content}}$$

$$\text{Util total cache network} = \sum_{n=0}^{\text{num_cache_nodes}} \text{util_cache_node}_n * \frac{\text{num_requests_at_cache_node}_n}{\text{total_num_requests_at_all_nodes}}$$

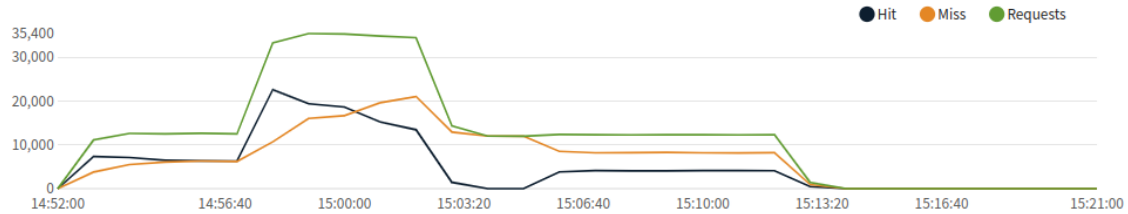
Graph Time

Initial Results

Experimental Results

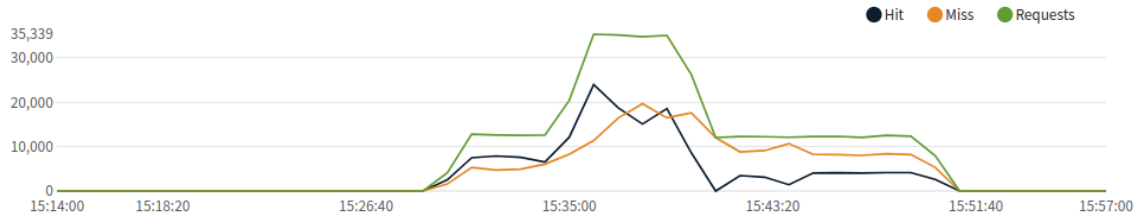
VC1 HIT AND MISS

Varnish Server: agentvc1 - last 30 mins (1 min interval)



VC1 HIT AND MISS

Varnish Server: agentvc1 - last 30 mins (1 min interval)



What did we learn and what will do next?

Conclusions and Future Work

Conclusions

- Many challenges await
 - Fitness
 - Encoding
- Case Study has been a success
- Simulation Model or Digital Twin will be necessary

Future Work

- Expand the implementation
 - Meta and Hyper levels
- Develop a more complex website
 - Diverse content types
- Build a more generic encoding
- Fitness Function improvements

So can you see it working?

Funny you should ask...

Thanks for listening!
Any questions?