

Replatforming Layer7 APIM and APIOPS Automation

Carlos Pimentel
Senior Cloud Engineer – William Hill

Vince Baker
Principal Architect - Apiida



Bigger, Stronger, Better Together

apiida

Vince Baker

*William***HILL**

Telefonica
telecom

oberthur
TECHNOLOGIES
THE M COMPANY

EGC
Telecom Integration

BT

MANSION

worldpay

upc

RBS
The Royal Bank of Scotland

Carlos Pimentel

About

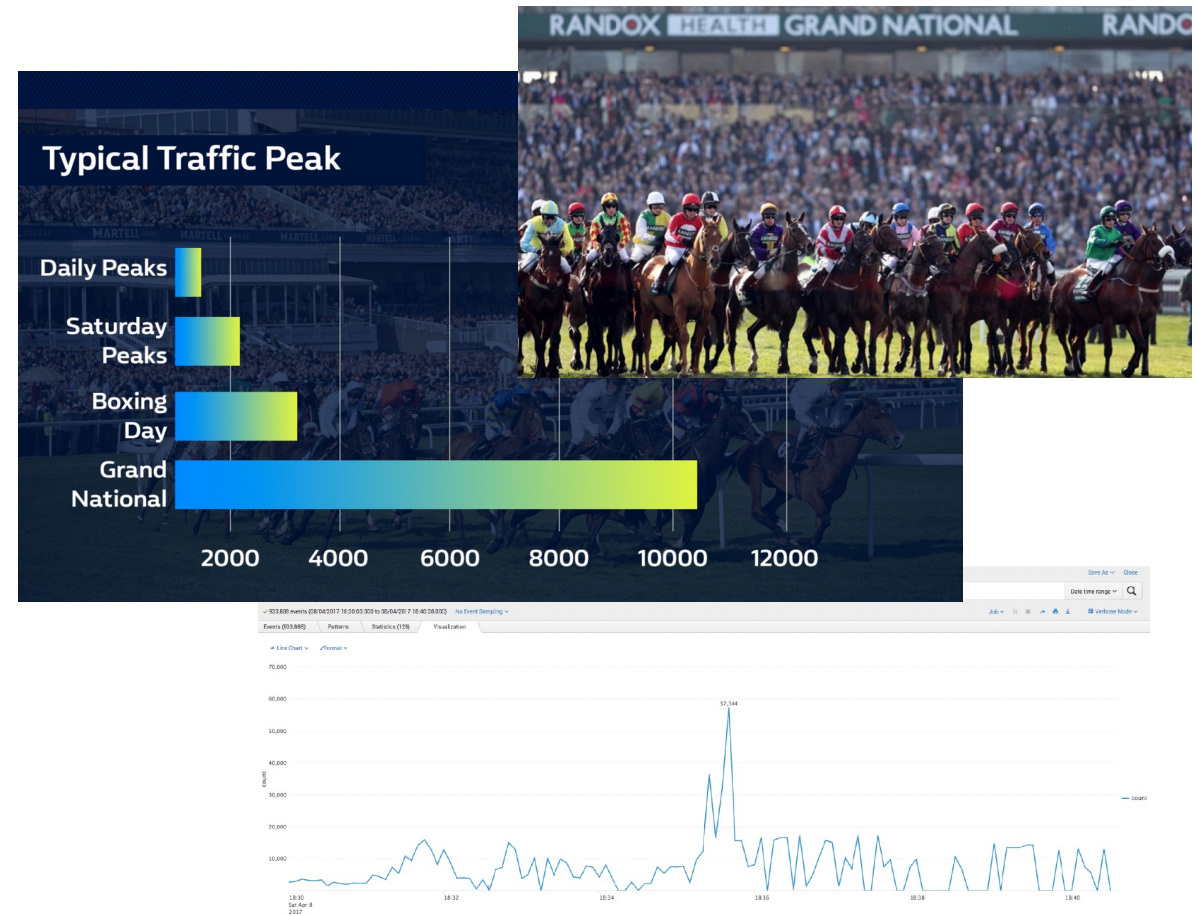
WILLIAM HILL



- Bookmaker – Retail and online
- Founded in 1934 in Leeds
- Currently over 16,000 employees
- Retail Offices throughout UK
- Online Offices in 8 different countries
- William Hill uses many CA products




History of API gateway platform at WH

- Around 10 years using Layer7 APIM at WH
- Started with 3 Virtual Appliance gateways, then 7 (geographically split into 2 clusters). Increased to 56 gateways in production to handle key events.
- Hard to maintain patching and policies change.
- Unsupported ESM and GMU was highly inefficient.
- Many Classic Portals with problematic custom sync.
- Mix of badly written v1 APIs and well written v2's (using a framework).
- Performance was good - Proved itself with the GN horse race event in 2017 with a 11k req/s spike.

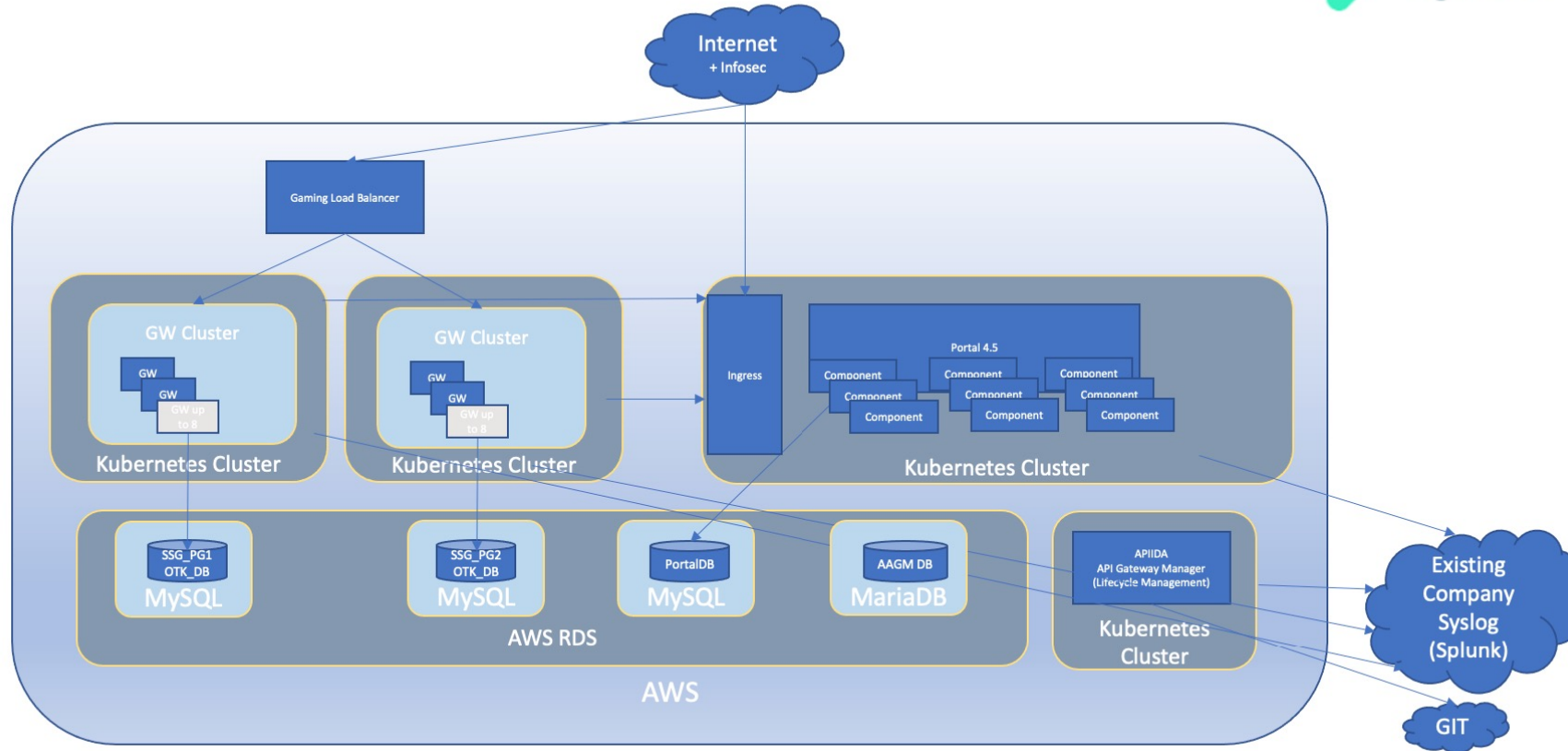


- Primary requirement - Remove all WH API systems from the existing datacentres and move to AWS.
- Allow scaling for key events in the gaming calendar
 - Automated scaling of gateways
- Ease of API version manager  apiida veen environments
 - Version Control using GIT.
- Whilst we're at it.....
 - Upgrade Gateways to v10
 - Upgrade the Developer Portal to v4.x (now v5.x)
 - Preserve customer/consumer API Keys/Secrets
 - Move away from ESM and streamline the API lifecycle process
 - Split the Gaming and Sports business verticals
 - Shiny new framework designed around cloud  structure (v3 Framework/apiida API Service Manager)



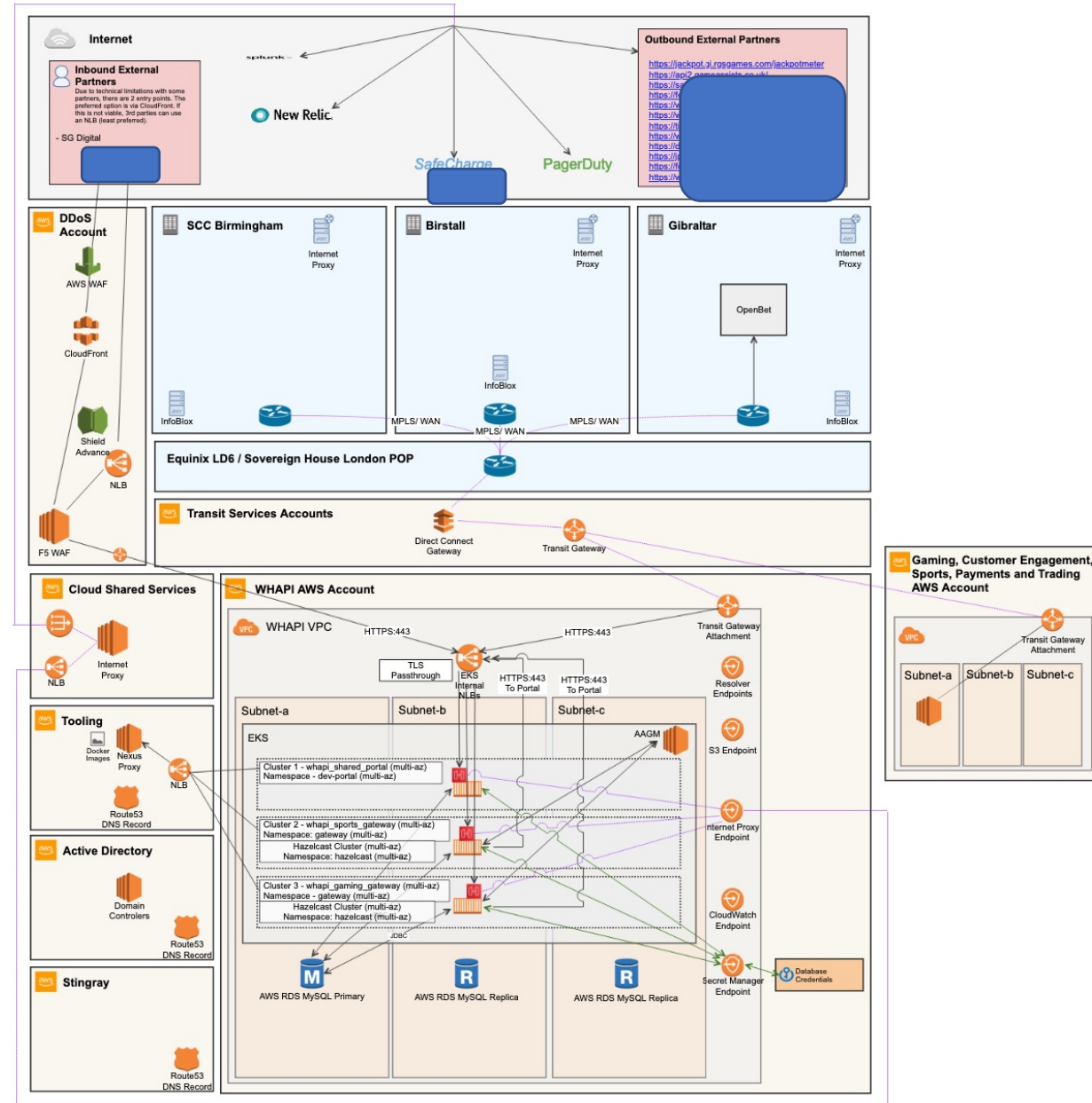
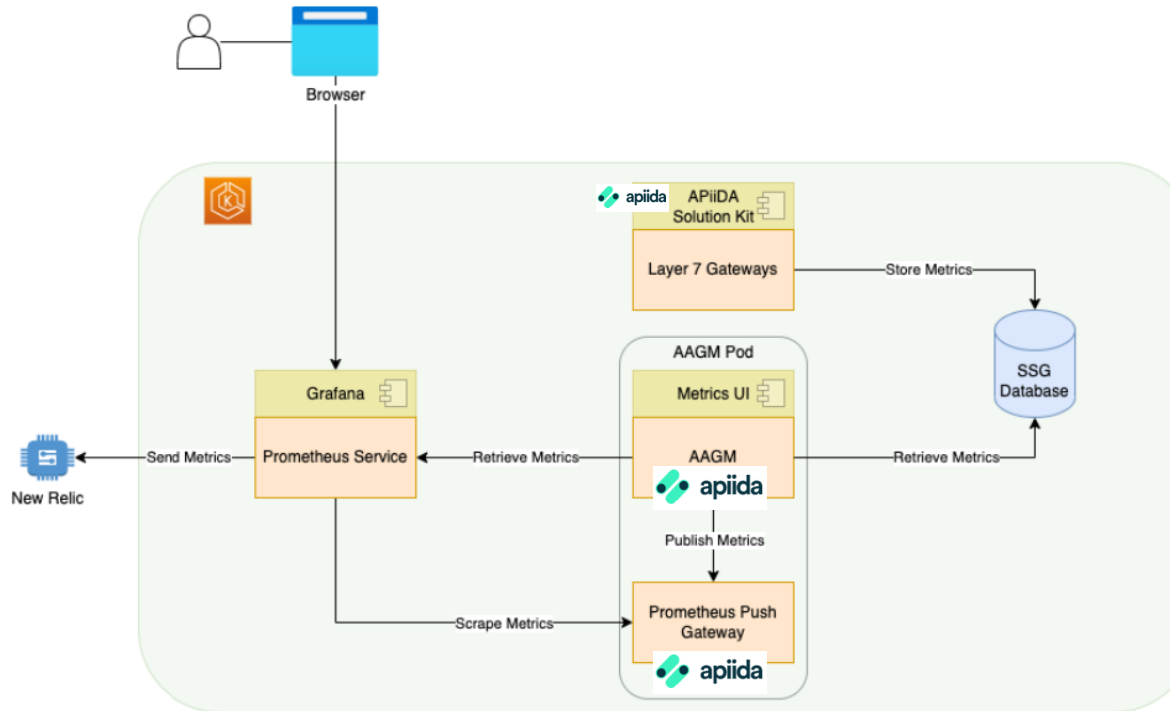
- AWS Native Gateway & Kong were considered.
 - AWS “per request” cost was too high.
 - Didn’t provide enough orchestration - Middleware Lambda code needed.
- Choices of L7 form factor considered
 - Docker on AMI nodes,
 - VMC (VMWare appliances in AWS)
 - AMI images
 - Containers managed by Kubernetes (the winner).
- 4.5 Developer Portal (Now at 5.1)
 - Migrate Prod applications/keys using migration tool.
- V10.x Gateway
- API Lifecycle (API Gateway Manager)  **apiida**
 - WH and apiida has been working closely together to help develop AAGM as the WH gateway setup is challenging and great for finding new feature requirements for AAGM.
- apiida API Service Manager (Gateway framework)

Proposed Infrastructure (per environment)



Final Solution





- New Framework was implemented (AASM)
- Move solution to AWS using EKS
- Use of external hazelcast and database.
- Fast deployment using AAGM
- Gathering metrics with AAGM
 - Feeding into Prometheus



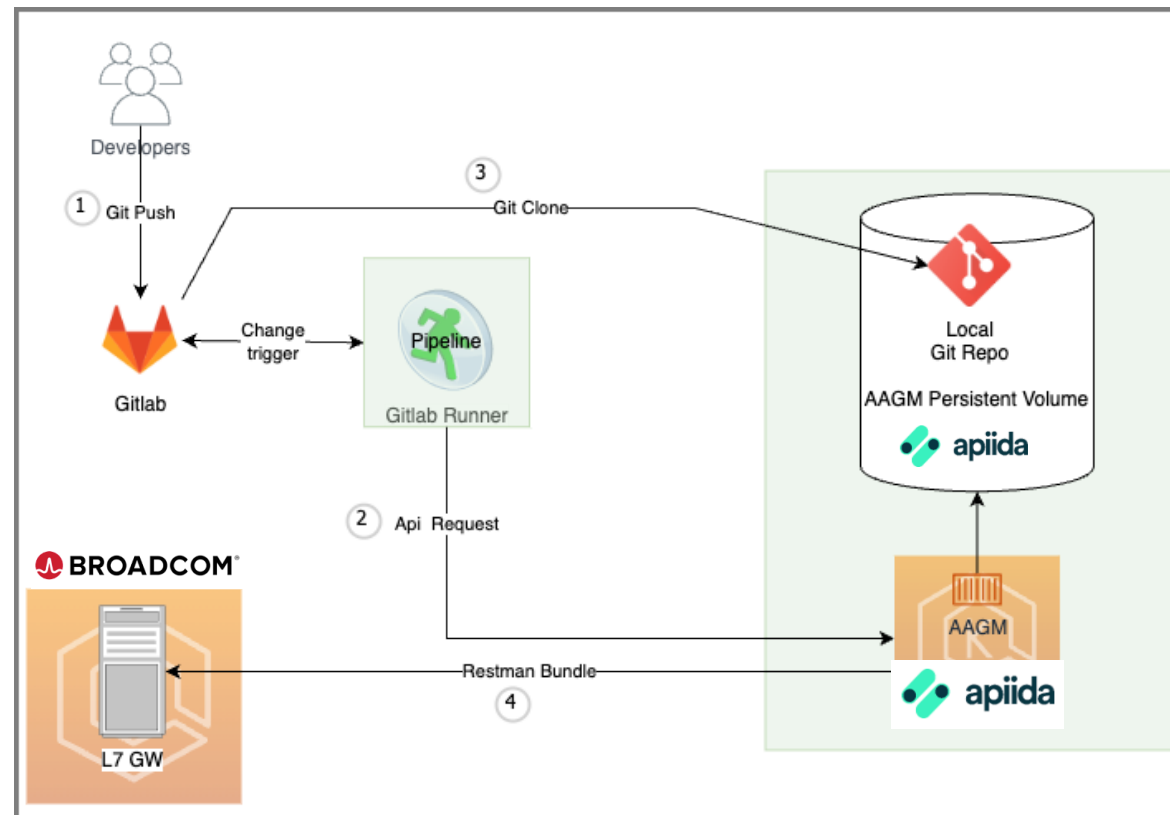
- Gateway – Supportable version. No need for separate v10 upgrade of current deployment.
- Highly scalable (To deal with spikes in traffic)
- Ease of patching and upgrades for Gateway and Portal
- Portal
 - Multi Cluster
 - Replace flaky custom API Portal sync scripts
- Faster policy dev and service admin with improved framework.
- AWS EKS solution could be migrated to other cloud providers if ever needed.

- Initial problem with Framework relying on Gateway “Names” which were dynamic. – GW didn’t know its identity and therefore it’s routing, CWP’s, Backends, routing, logging etc. Fixed with cloud aware framework.
- Early 4.x Portal didn’t have API Plans.
- Portal – Quotas are no longer reset hourly. Minor issue.
- Portal data had many issues during 3.5 -> 4.5 migration.
- Getting GW metrics into NewRelic



- EKS deployed with Terraform into AWS 
- API Gateways and Portal deployed with Kustomize 
- EKS nodes - 32 CPU, 64GB RAM – GW pods 8 CPU and 8 RAM (6GB Java)
- Istio for load balacing and also Blue-Green deployment 
- Gitlab Pipelines 
- Separate portals for Prod and NonProd
- Currently running a base of 24 gateways in sports and 4 in gaming.
- Scalable to 96 in sports and 32 in gaming without extra deployment config.

- Automated deployment of changes using AAGM APIs with GitLab (CI/CD).
- Automatic LB traffic masking, smoke test, then unmasking if tests pass.
- Event based autoscaling.



That's all folks!

QUESTIONS?

Thank you for for being a
great audience, for listening
to our story and joining in.