

TrackHub Registry In Production

03/02/2016

EMBL-EBI



EMBL – European Bioinformatics Institute
Wellcome Trust Genome Campus
Hinxton, Cambridge, CB10 1SD, UK



EMBL-EBI



Update

<https://beta.trackhubregistry.org>

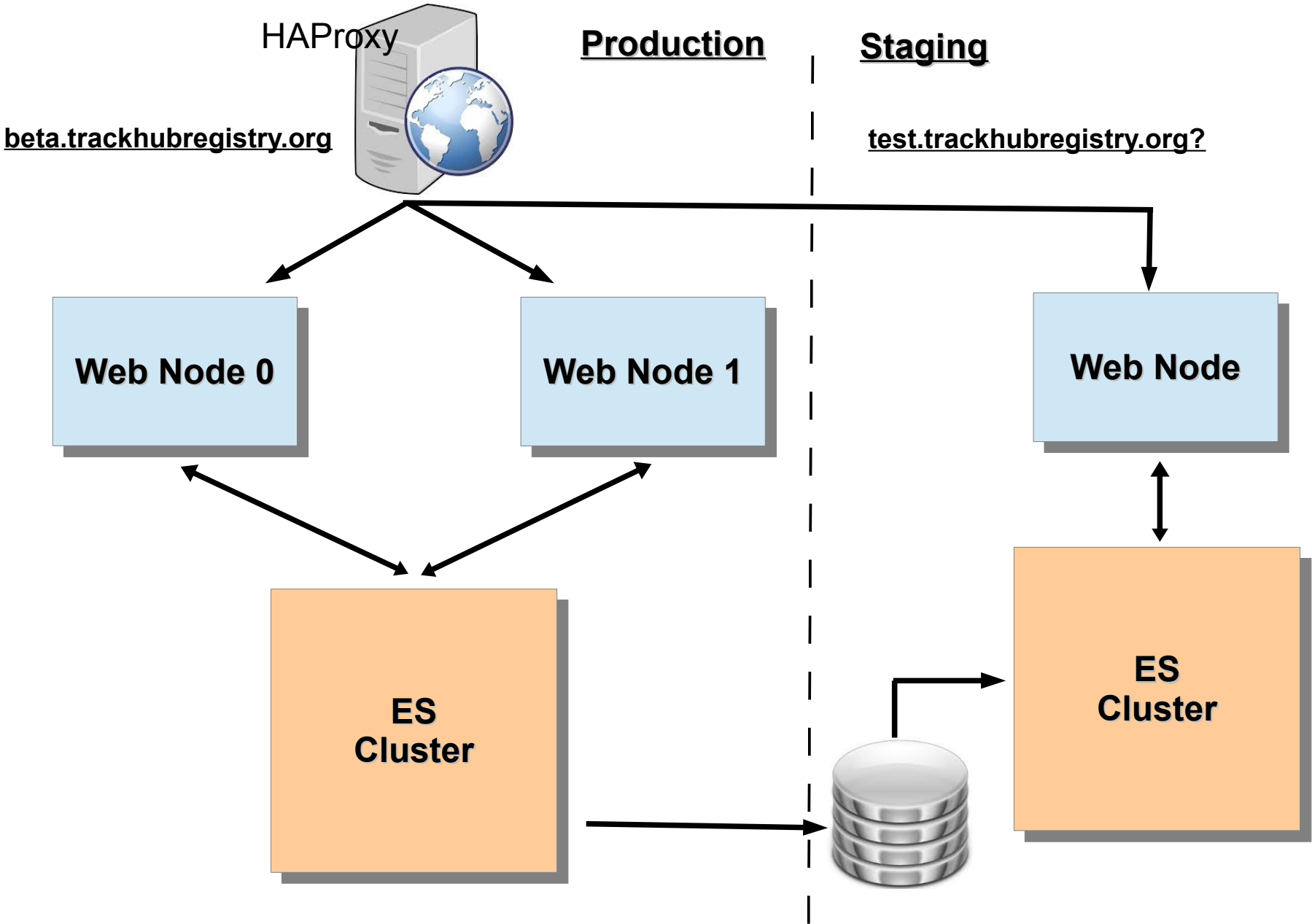
- 4 registered providers
- 799 submitted track hubs
 - >700 plants
 - 16 human
 - 5 mouse
 - 5 zebrafish

Track Submission

- Non plant track hubs
 - submitted wo author consent, contact and transfer ownership!
 - list from e! public page
http://www.ensembl.org/info/website/public_trackhubs.html
 - some were not possible

E! public hubs

- 21 hubs, 14 submitted, 7 had problems
- Not found
 - Deutsches Epigenome Programm (DEEP)
 - UMassMed H3K4me3 ChIP-seq data
- 503: service temporarily unavailable
 - CREST IHEC Epigenome Project Hub
- 504: gateway time out
 - DNA Methylation Hundreds of analyzed methylomes from bisulfite sequencing data
- Timed out
 - Sense/antisense gene/exon expression using Affymetrix exon array from South Dakota State University
- Error: chars not mapping to Unicode
 - EDACC hosted Roadmap Epigenomics Hub
- Error: parent track miRNA_expression is missing
 - Broad Improved Canine Annotation v1



HAProxy

- A single shared https based frontend
 - use crt-list option to configure multiple certs

/etc/haproxy/haproxy.cfg

```
frontend ft_https_in
  bind 0.0.0.0:443 ssl crt-list /etc/haproxy/certs
  reqadd X-Forwarded-Proto:\ https

  acl rest hdr(host) -i rest.ensembl.org
  use_backend bk_rest if rest

  # acl grch37_rest hdr(host) -i grch37.rest.ensembl.org
  # use_backend bk_grch37_rest if grch37_rest

  acl thregistry hdr_end(host) -i trackhubregistry.org
  use_backend bk_thregistry if thregistry
```

/etc/haproxy/certs

```
/etc/haproxy/trackhubregistry.pem *trackhubregistry.org
/etc/haproxy/haproxy-sanger-CA.pem rest.ensembl.org
```

The Backend (Prod)

- 2 s1.large instances
 - RAM 8GB, 4 VCPU, Disk 60GB
 - thregistry_web_0|1: 192.168.0.89|93
 - Access key: trackhub.pem
 - Ubuntu 14.04 (kernel 3.13.0-48 SMP)
- Each one runs the web site and REST API
 - Starman on port 5000
 - 5 workers

Backend code

- TrackHub registry (Catalyst)
 - /home/ubuntu/src/trackhub-registry/src/Registry (\${REGISTRY_HOME})
- Dependencies in /home/ubuntu/src
 - various ensembl repos
 - includes ensemblgenomes-api

Backend operations

- System monitoring
 - Installed Sysstat
 - `$ sar -f /var/log/sysstat/saXX`
- Various perl scripts in:
 - `${REGISTRY_HOME}/scripts/production`
- Access/Error logs:
 - `${HOME}/perl5/registry.(access|error).log`
- No kibana/logstash at the moment

Production scripts

(in \$REGISTRY_HOME/scripts/production)

- Start|Stop the server
 - *(start|stop)_server.sh*
 - config in \$REGISTRY_HOME/conf/production/registry.conf
- Reset the cluster (danger!)
 - *init_cluster.pl*
 - option controls which cluster (prod|staging)
 - read .initrc by default on the same directory

```

<TrackHub>
  <schema>
    default v1.0
    v1.0 __path_to(root/static/trackhub/json_schema_1_0.json)__
  </schema>
</TrackHub>

<Model::Search>
  <trackhub>
    index trackhubs
    type trackdb
  </trackhub>
  <user>
    index users
    type user
  </user>
  <report>
    index reports
    type report
  </report>
  max_requests 10000
  nodes 192.168.0.96:9200
  nodes 192.168.0.97:9200
  request_timeout 30
</Model::Search>

<GenomeCollection>
  assembly_set_file = __path_to(root/static/gc_assembly_set.json.gz)__
</GenomeCollection>

<Model::GenomeAssemblyDB>
  schema_class Registry::GenomeAssembly::Schema
  traits Caching
  <connect_info>
    dsn DBI:Oracle:host=ora-vm5-003.ebi.ac.uk:sid=ETAPRO:port=1571
    user gc_reader
    password reader
  </connect_info>
</Model::GenomeAssemblyDB>

<Plugin::Authentication>
  use_session 1
  <realms>
    <web>
      <store>
        index users
        store_user_class Catalyst::Authentication::Store::ElasticSearch::User
        nodes 192.168.0.96:9200
        nodes 192.168.0.97:9200
        class ElasticSearch
        type user
      </store>
      <credential>
        class Password
        password_field password
        password_type clear
        username_field username
      </credential>
    </web>
  </realms>
</Plugin::Authentication>

```

```
[cluster_prod]
name : thregistry
nodes: 192.168.0.96:9200
# add other nodes here, will be assigned to the same nodes arrayref
nodes: 192.168.0.97:9200

[cluster_staging]
name : thregistry_staging
nodes : 192.168.0.100:9200

[trackhubs]
index  : trackhubs_v1
alias  : trackhubs
type   : trackdb
mapping : /home/ubuntu/src/trackhub-registry/docs/trackhub-schema/v1.0/examples/trackhub_mappings.json
# could specify some settings with index API,
# see https://www.elastic.co/guide/en/elasticsearch/reference/current/indices-update-settings.html
number_of_shards : 3
number_of_replicas: 1

[users]
index      : users_v1
alias      : users
type       : user
mapping    : /home/ubuntu/src/trackhub-registry/docs/trackhub-schema/v1.0/examples/authentication_mappings.json
admin_name : admin
admin_pass :
number_of_shards : 1
number_of_replicas: 1

[reports]
index      : reports_v1
alias      : reports
type       : report
number_of_shards : 1
number_of_replicas: 1

[repository]
name      : backup
type      : fs
location  : /mnt/backup
remote_location : /mnt/es_snapshots
```

Production scripts (2)

- JSONise Genome Assembly DB table
 - *dump_genome_assembly_set.pl*
 - run periodically
- Monitor registered hubs
 - *update_trackdb_status.pl*
 - should run as cron job but not until mail can be sent

ES Cluster (Prod)

- 2 m1.EBI-Ensembl16 instances
 - RAM 16GB, 4 VCPUs, Disk 120GB
 - thregistry_cluster_0|1: 192.168.0.96|97
 - Access key: trackhub.pem
 - Ubuntu 14.04 (kernel 3.13.0-48 SMP)
- Cluster: **thregistry**, no elected master
- Run elasticsearch 1.7.3 (from deb)

ES node configuration

- Files:
 - **/etc/elasticsearch/elasticsearch.yml** (not a lot there, just cluster name, path.repo and memory options)
 - **/etc/default/elasticsearch** (configure ES init service, i.e. HEAP size 8GB, max open files 65535, locked memory)
 - **/usr/lib/systemd/system/elasticsearch.service** (override memory options elsewhere)
- Logs written to **/var/log/elasticsearch**
- Index written to **/var/lib/elasticsearch**
 - 120GB with two RAID0 disks (/dev/vdb1, /dev/vdc1)
 - pros: high R/W throughput
 - cons: no redundancy/fault tolerance, might be tricky adding new disk
 - /etc/mdadm/mdadm.conf
- Remotely mount staging server **/mnt/backup** under **/mnt/es_snapshots**

Indices et al.

Index	Alias	Type	Shards	Replicas
trackhubs_v1	trackhubs	trackdb	3	1
users_v1	users	user	1	1
reports_v1	reports	report	1	1

ES node operations

- System monitoring
 - `$ sar -f /var/log/sysstat/saXX`
- Start/Stop/Restart ES
 - `$ sudo elasticsearch start|stop|restart`
- TODO: cluster monitoring
 - use some plugins, e.g. Marvel (free in dev)

Staging server

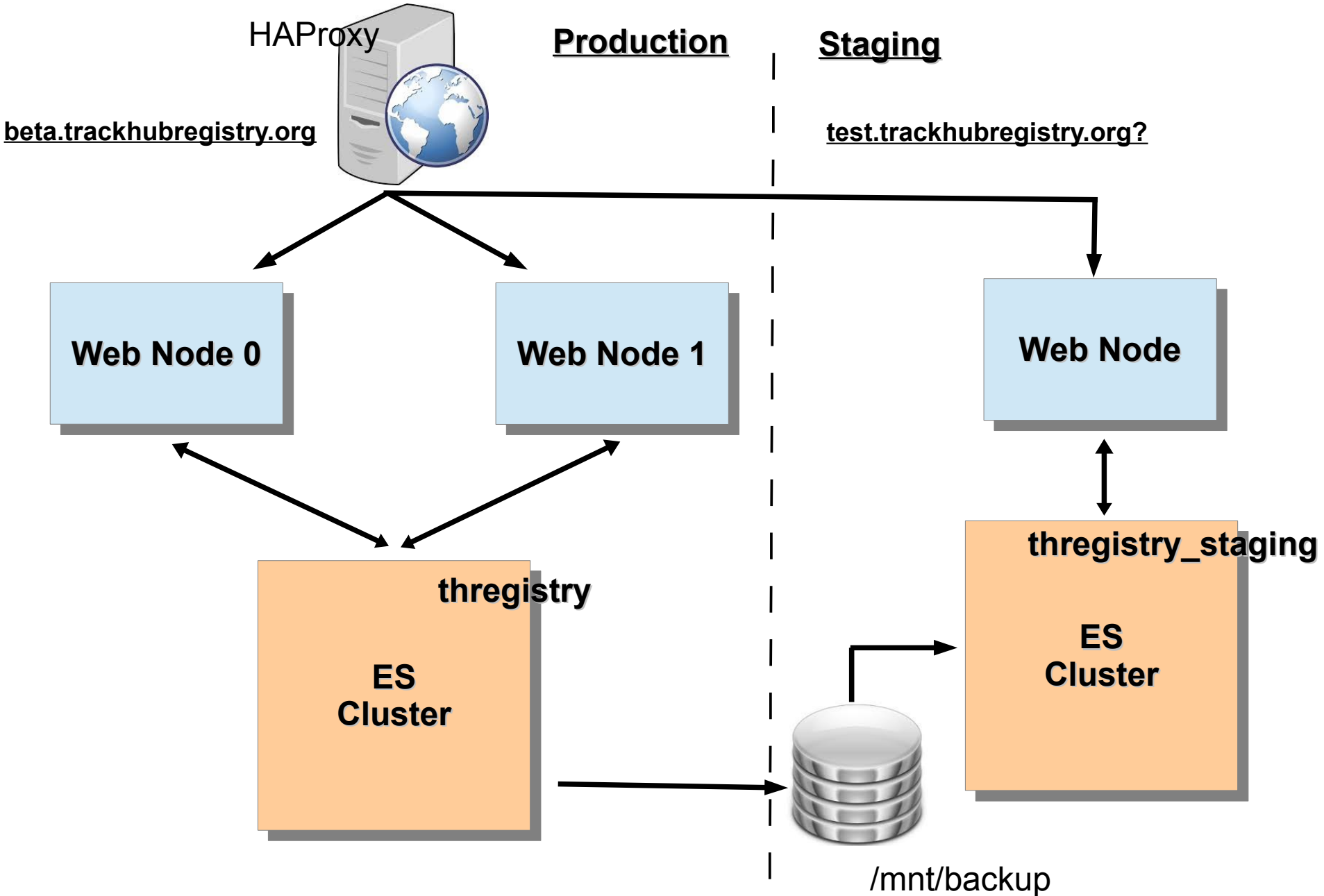
- Purpose
 - prepare new data (re-index)
 - test new developments of the interface
 - test recovery from snapshots
- 1 s1.small instance
 - RAM 2GB, 2 VCPUs, Disk 20GB
 - **thregistry_staging_backup**: 192.168.0.100
 - Access key: trackhub.pem
 - Ubuntu 14.04 (kernel 3.13.0-48 SMP)

Backend operations

- System monitoring
 - Installed Sysstat
 - `$ sar -f /var/log/sysstat/saXX`
- Various perl scripts in:
 - `${REGISTRY_HOME}/scripts/production`
- Access/Error logs:
 - `${HOME}/perl5/registry.(access|error).log`

ES node configuration

- Cluster: **thregistry_staging**
 - elasticsearch 1.7.3 (from deb)
- Files:
 - **/etc/elasticsearch/elasticsearch.yml**
 - **/etc/default/elasticsearch** (HEAP size 1G)
 - **/usr/lib/systemd/system/elasticsearch.service**
- Logs written to **/var/log/elasticsearch**
- Index written to **/var/lib/elasticsearch**
 - 120GB with two RAID0 disks (/dev/vdb1, /dev/vdc1)
 - /etc/mdadm/mdadm.conf
- Mount 120GB disk **/mnt/backup** (/dev/vdd1) to store snapshots



Recovery

shards have replicas, but you never know ...

- takes regular snapshots of the data
- tests recovery from the latest
- sends alert (sending emails?)

Script still to test/complete, will be at

- `${REGISTRY_HOME}/scripts/production/`

New data/development

- Scenario:
 - implement a change in the interface
 - might require re-indexing of the data
- How to re-create the index and test it's working before deploying to the prod cluster?