New Developments in Search at NCBI

Querying Feature Annotations

&

High Availability Solr Stack in AWS

Overview

Querying Feature Annotations

High Availability Solr Stack in AWS

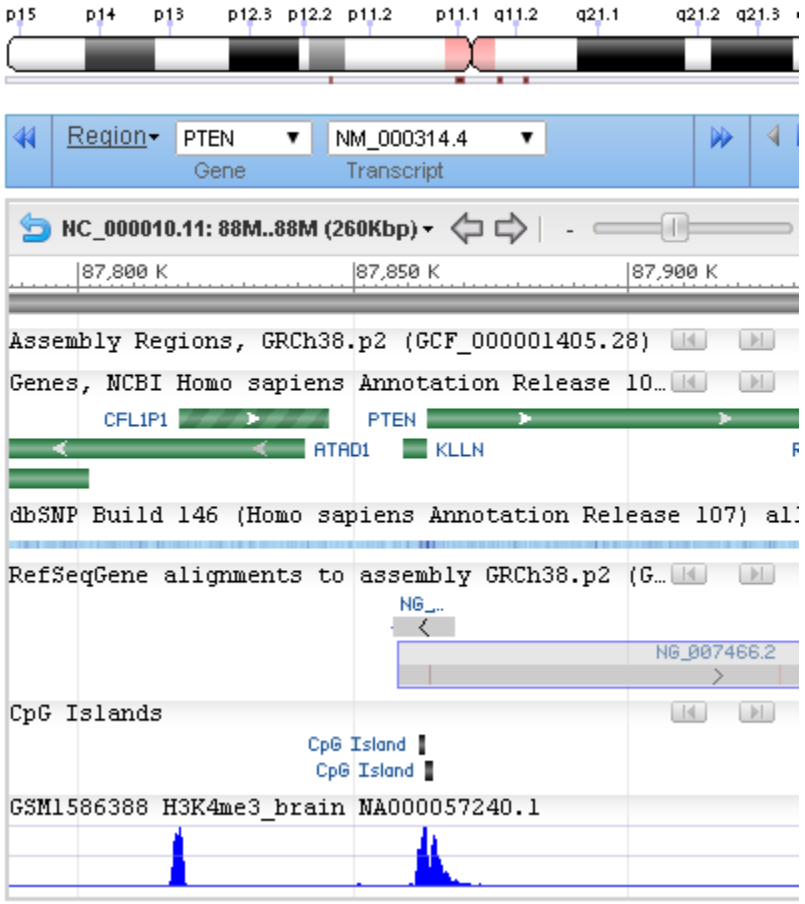
Querying Feature Annotations

Feature Location Service

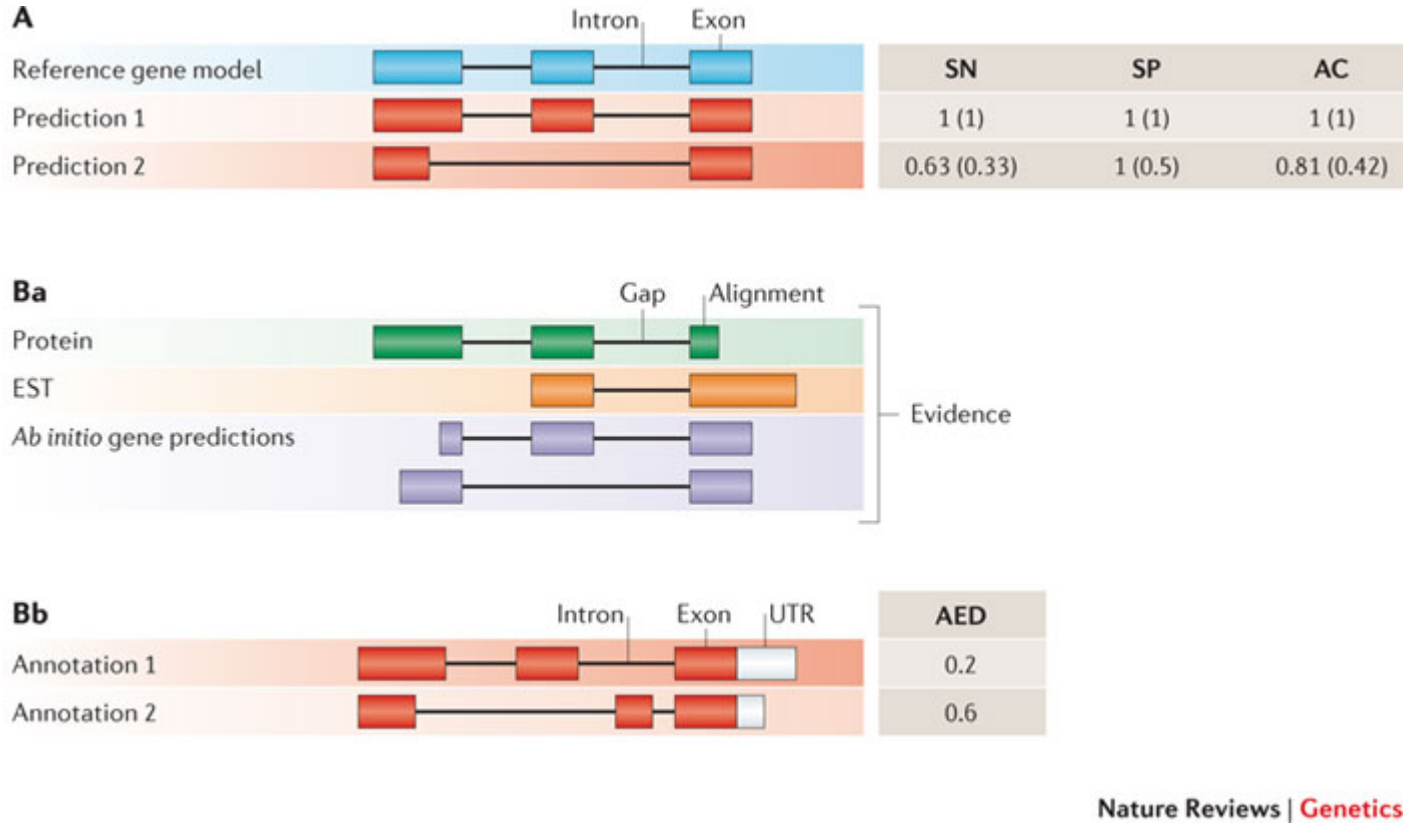
Team Lead: Valerie Schneider PhD
Software Engineers: Peter Meric, Cliff Clausen PhD

Feature Annotations

- Genes
  - Locus
  - RNA
  - Protein-coding region
- Variation
  - SNPs
  - Structural variants
- Clones

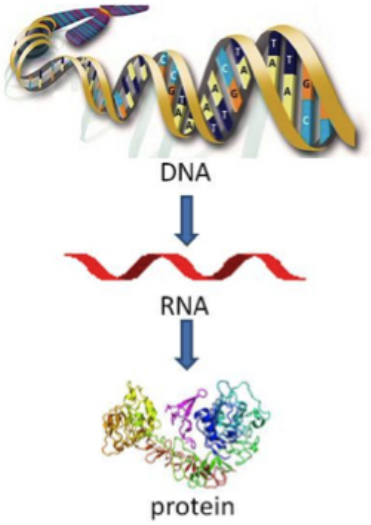


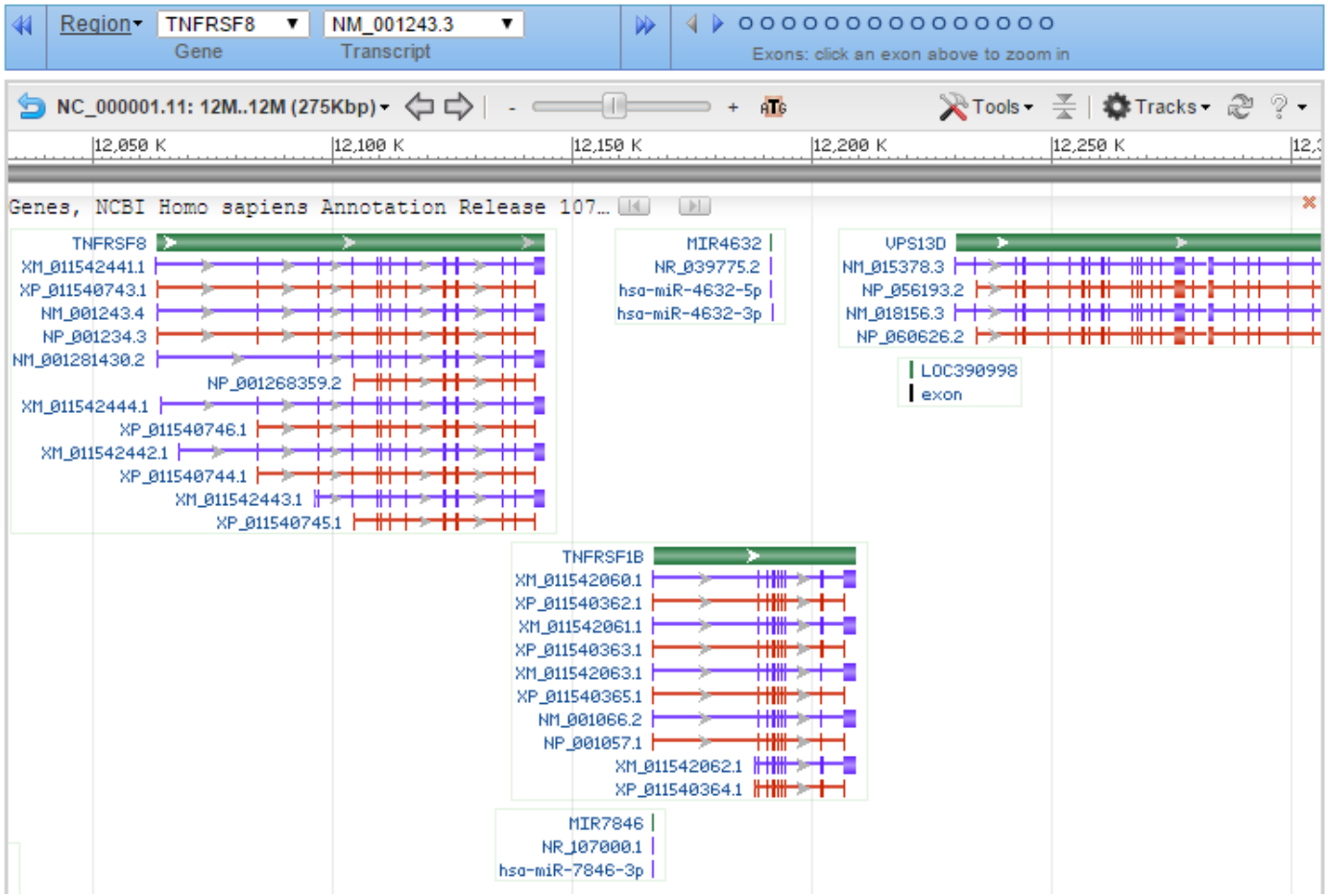
Genome annotation—Genes



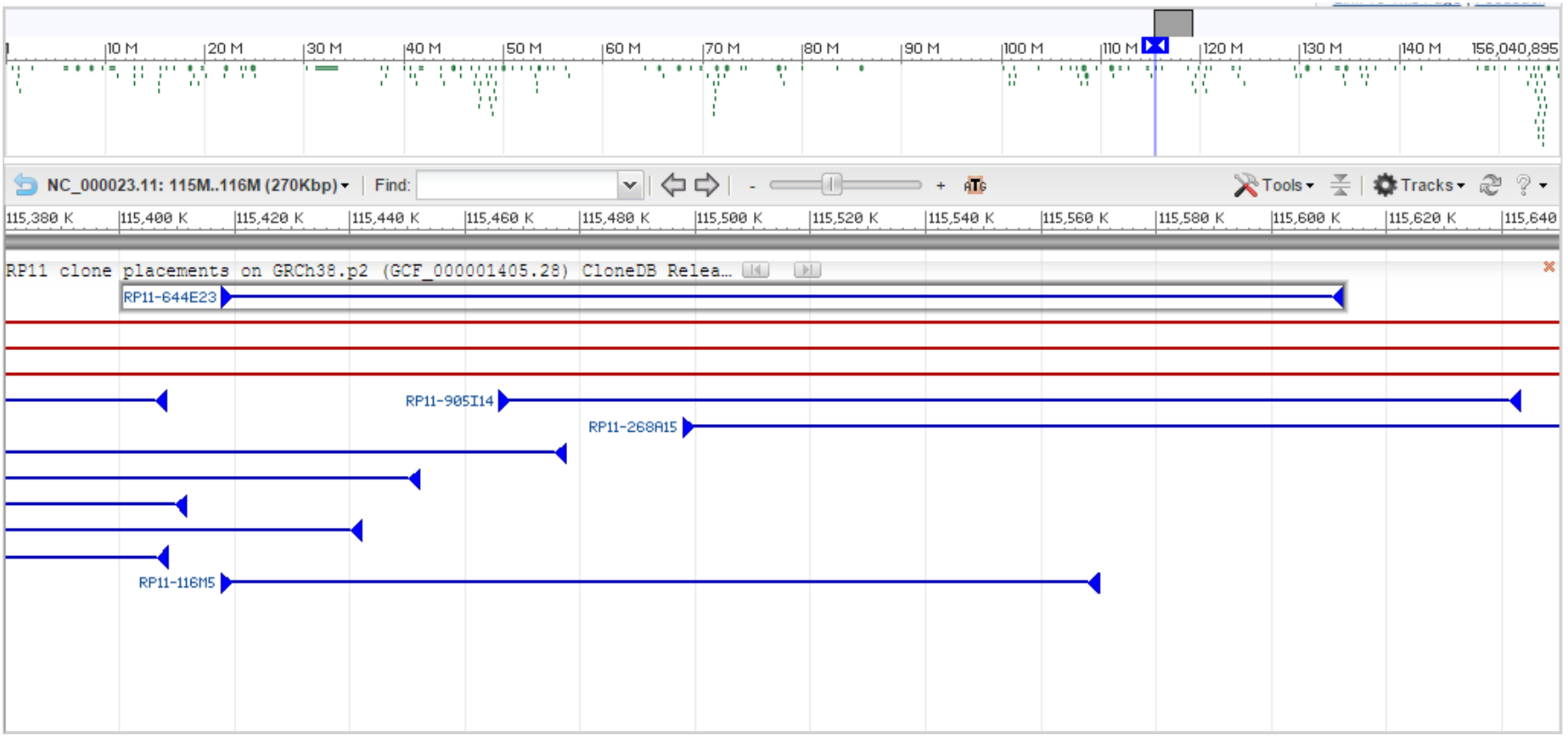
A beginner's guide to eukaryotic genome annotation. Yandell M, Ence D. Nat Rev Genet. 2012 Apr 18;13(5):329-42. doi: 10.1038/nrg3174.

Genome annotation—Genes

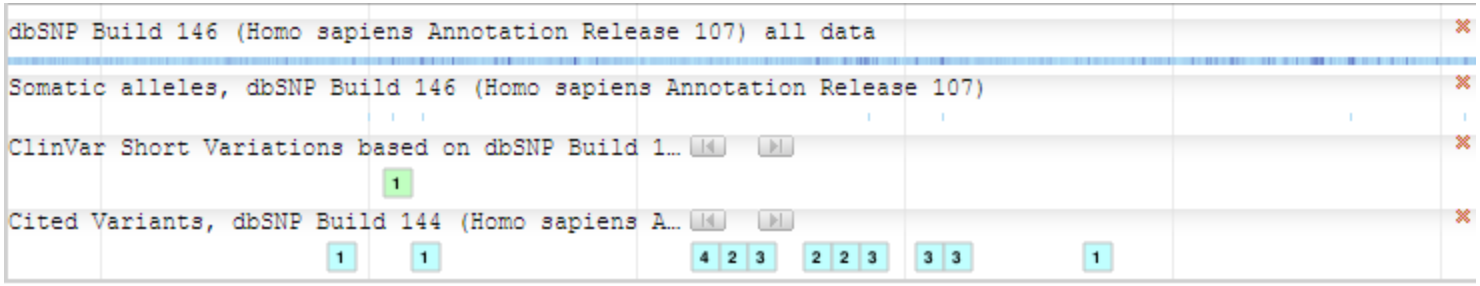


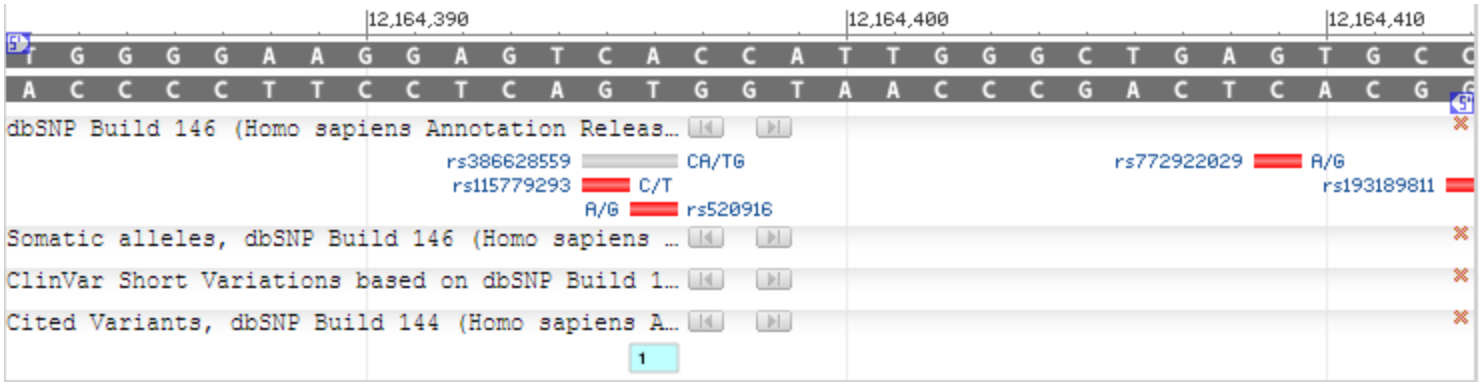


Genome annotation—Clones



Genome annotation—SNPs

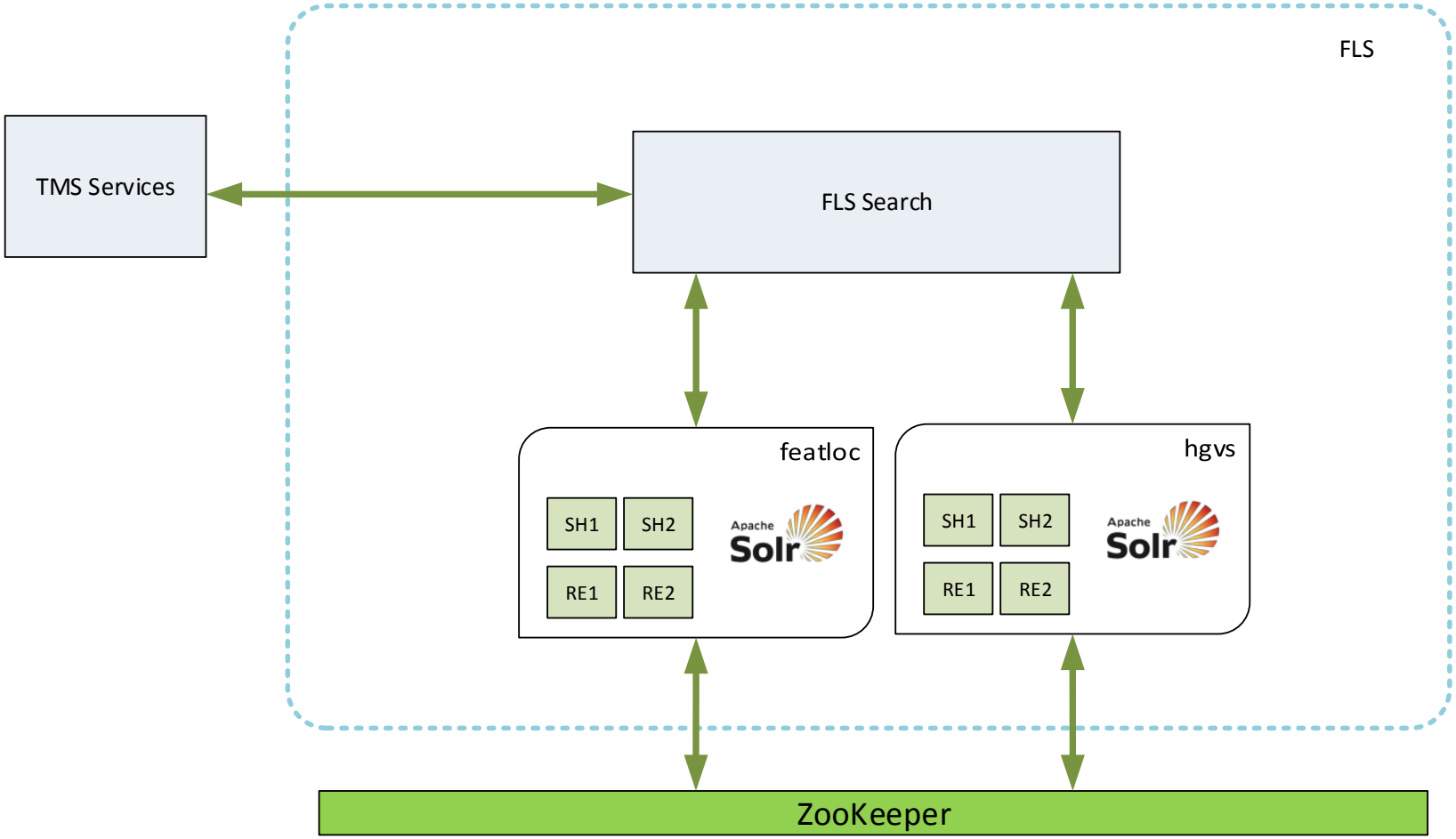




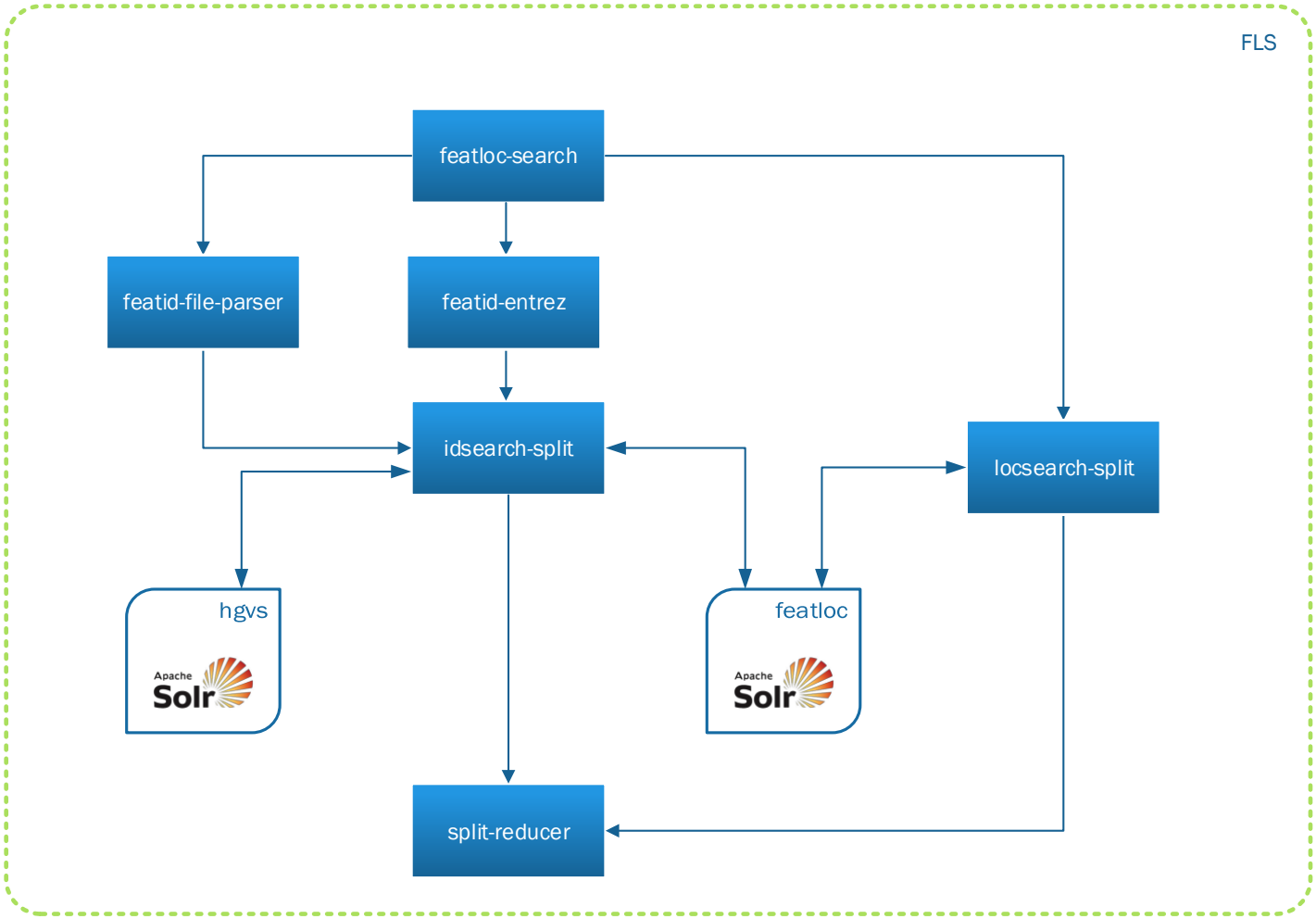
Why do we need FLS?

- Need to find annotations of interest
  - By location
  - By feature identifier
    - From publications
    - External queries from feature-specific resources eg. Gene
- To enable feature searching in NCBI genome browsers

FLS Architecture



FLS Search Services



Featloc schema

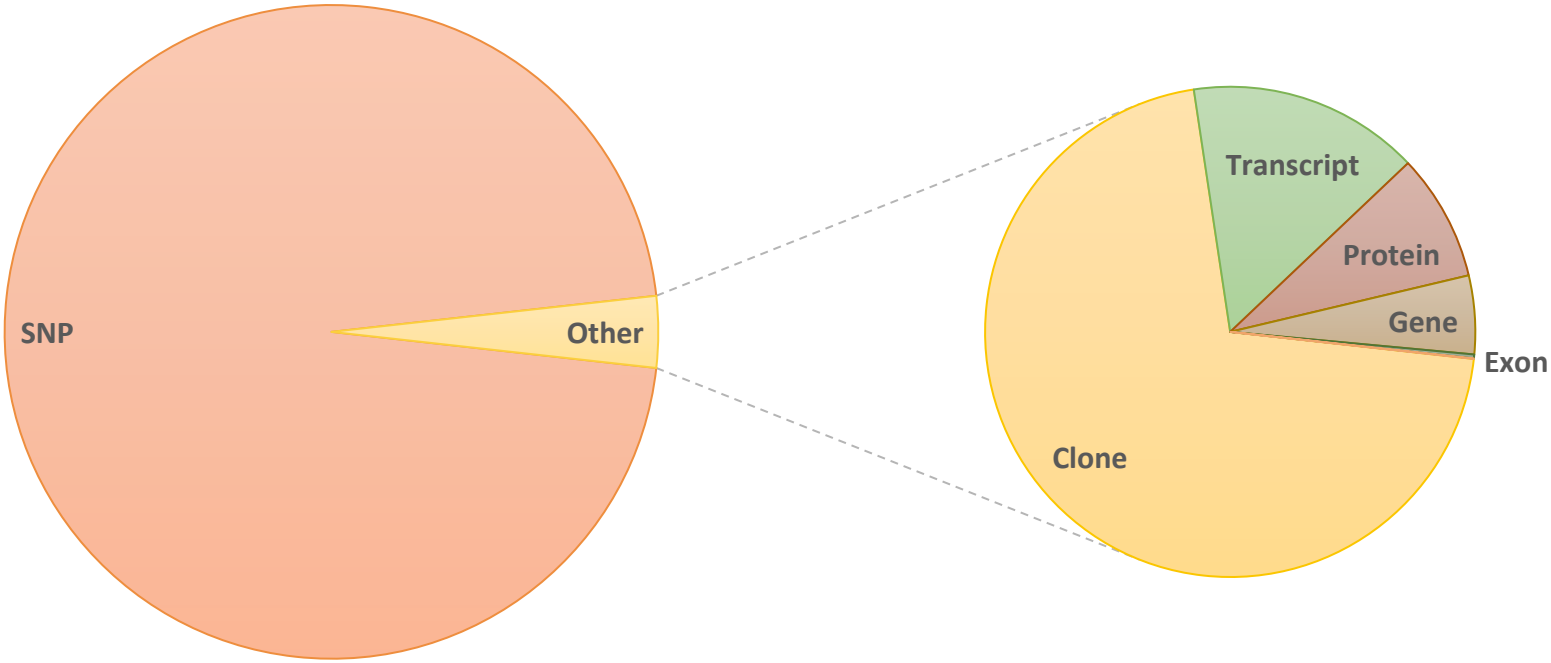
Field name	Type	Indexed	Stored	Multivalued	Required
feature_id	String	✔	✔	✔	✔
featuretype	String	✔	✔	✔	✔
accession	String	✔	✔		
fpl_id	String	✔	✔		
xref_fpl_id	String	✔		✔	
any_fpl_id	String	✔		✔	
symbol	String	✔	✔		
gi	Long	✔	✔		
start	Integer	✔	✔		✔
stop	Integer	✔	✔		✔
length	Integer	✔	✔		✔
strand	String	✔	✔		
track_id	String	✔	✔	✔	✔
seq_feat	BinaryField		✔		

Featloc Solr collection

- Indexes top-level assembly locations for features
- Feature tracks
  - Total possible scope: 24,500 tracks for 430 organisms
    - Human: 4,128
    - Mouse: 381
    - Cow: 165
    - Next 63 organisms: 51-150 each
    - All other organisms each account for 50 or fewer
- 316 data tracks currently indexed in featloc
- 330M records, 50GB

Featloc track content—GRCh38

155M features



FLS Service—Queries

- Locations (accession, start, stop)
- Feature ID list
  - User-generated
  - Query-generated
- Example feature IDs:
  - Gene ID
  - dbSNP ID
  - dbVar id
  - Sequence ID (accession, GI)
  - HGVS expression

FLS Service—Searching by Location

- sequence identifier
- location
  - overlapping range
  - start range, stop range
- length constraint
- order by location (start, length)

Solr—Searching by Location

- q constraints
  - overlapping range `stop:[lower TO *] AND start:[1 TO upper]`
  - start range, stop range `start:[A TO B]`
- fq constraints
  - length constraint `start:[A TO B]`

FLS Service—Searching by Feature ID

- no predefined limit for feature ID list
- results ordered by:
  - sequence of input feature IDs
  - unordered
  - location (accession, start, length)

Solr—Searching by Feature ID

- q constraints
  - *:*
- fq constraints
  - feature type
  - uses terms parser for feature ID list
    - up to 15K per query
    - uncached
    - avoids scoring

Solr—Result sets

- Placement info from stored fields:
  - FPL ID, accession, feature type, start, stop, strand, symbol
- ASN.1 seq-feat
  - compressed binary format, string-encoded
  - detailed feature information

HGVS collection

Document count	Type	Unique key	Size
755M	Feature ID string HGVS string	Feature ID, HGVS	38GB
150M	Feature ID string HGVS string, multivalued	Feature ID	29GB

- Nomenclature heavily used by medical community to describe variants and locations
- HGVS collection adds HGVS expression support to FLS
- Each HGVS expression indexed twice in same document
  - complete expression “NM_003159.2:c.1675C>T”
  - suffix string, omitting sequence ID “c.1675C>T”
- 755M unique (feature ID, HGVS) tuples

Overview

Querying Feature Annotations

High Availability Solr Stack in AWS

High Availability Solr Stack on Amazon Web Services

Team Lead: Grisha Starchenko
Software Engineers: Michael Kholodov, Georgy Khoroshavtsev, Vadim Miller, Maxim Osipov

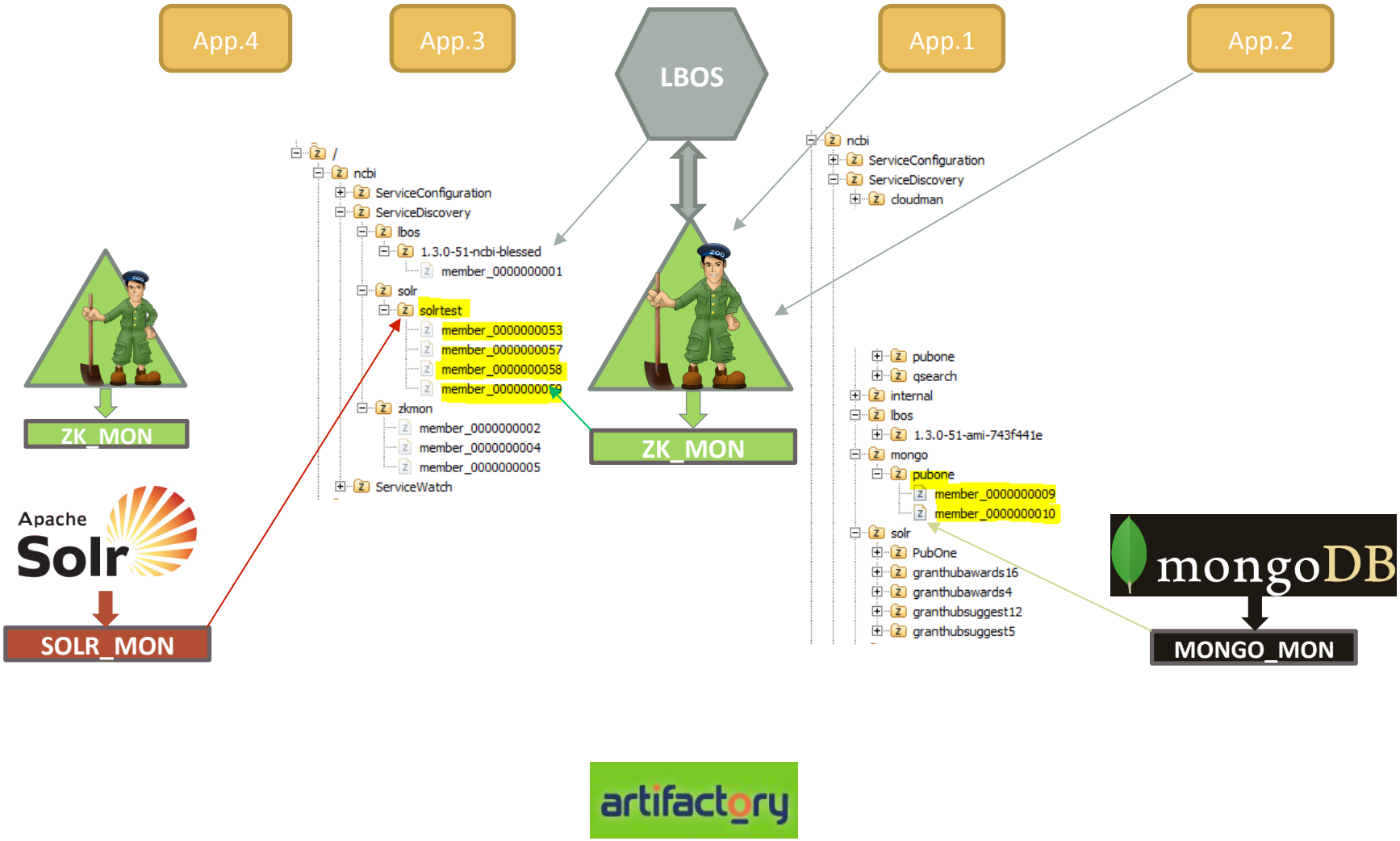
Why?

- PubOne (next generation PubMed search system)
- Scalable, distributed hardware
- Demand-driven resource allocation

Terms

- Service announcer
  - Zk_mon
  - Solr_mon
  - Mongo_mon
- LBOS – load balancer, dynamic configuration

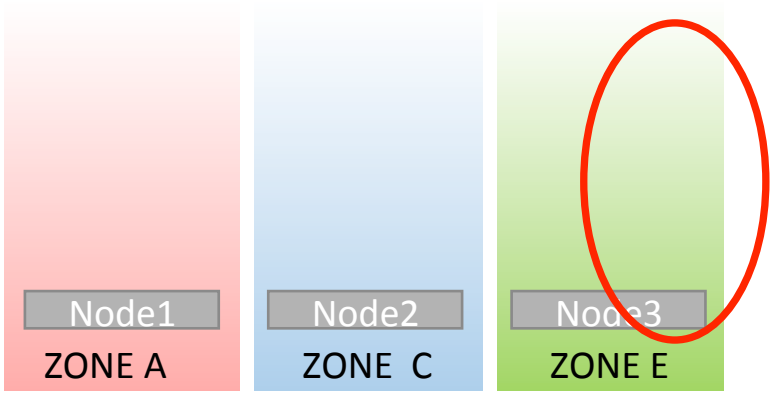
Interaction between services in the cloud environment



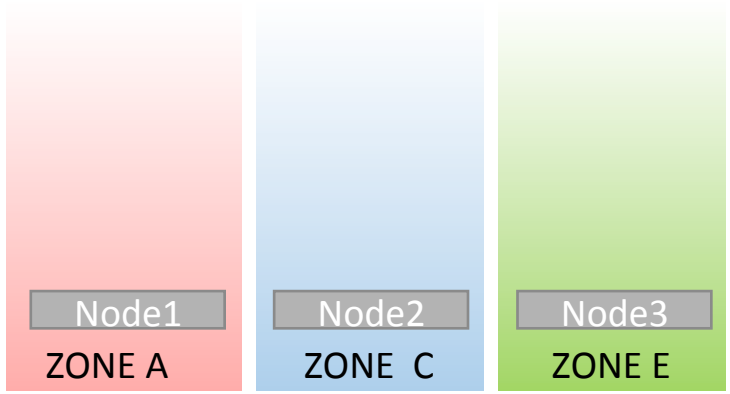
Shard-replica organization

INCORRECT

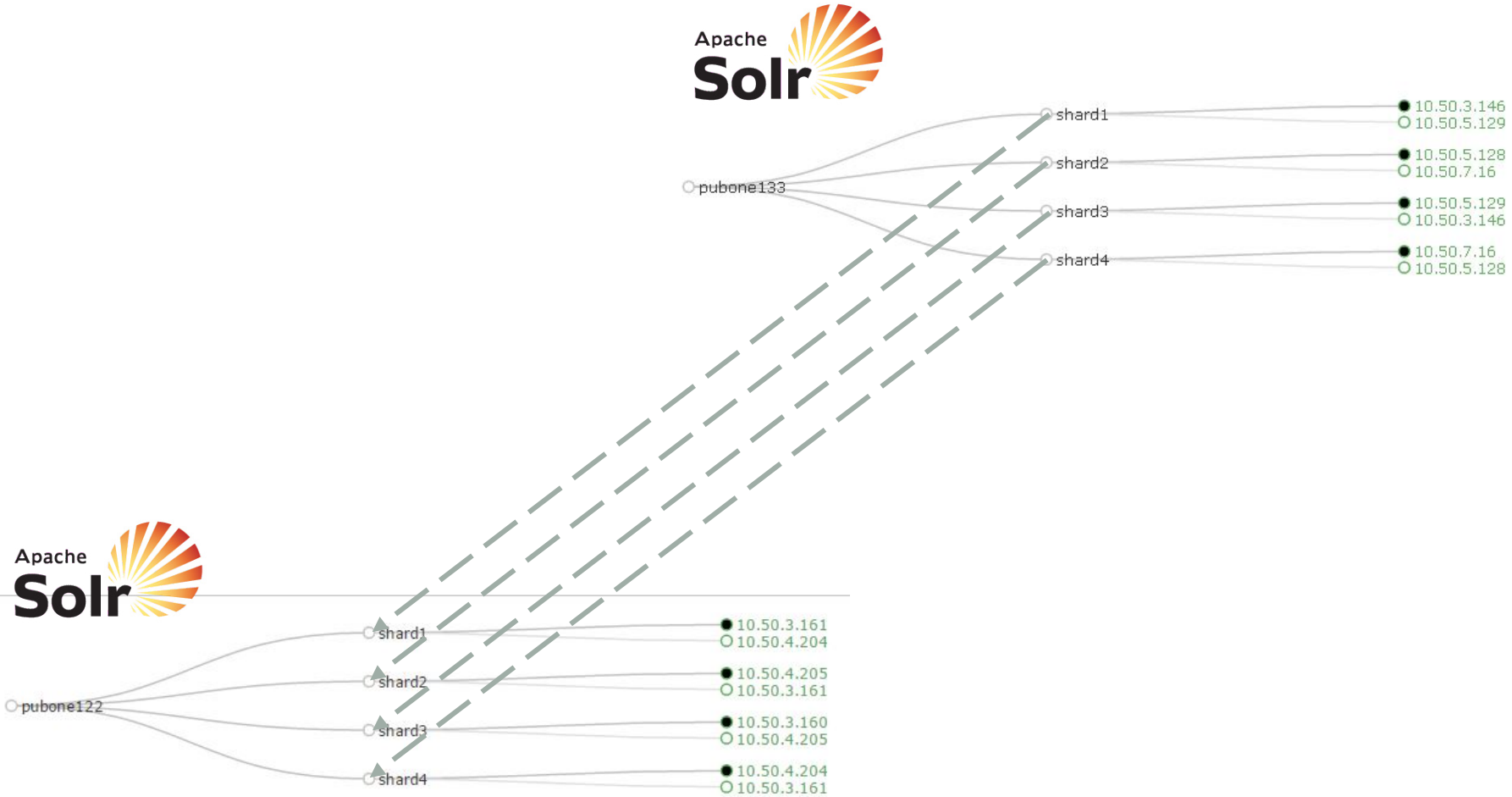
Both replicas of Shard_6 in same zone

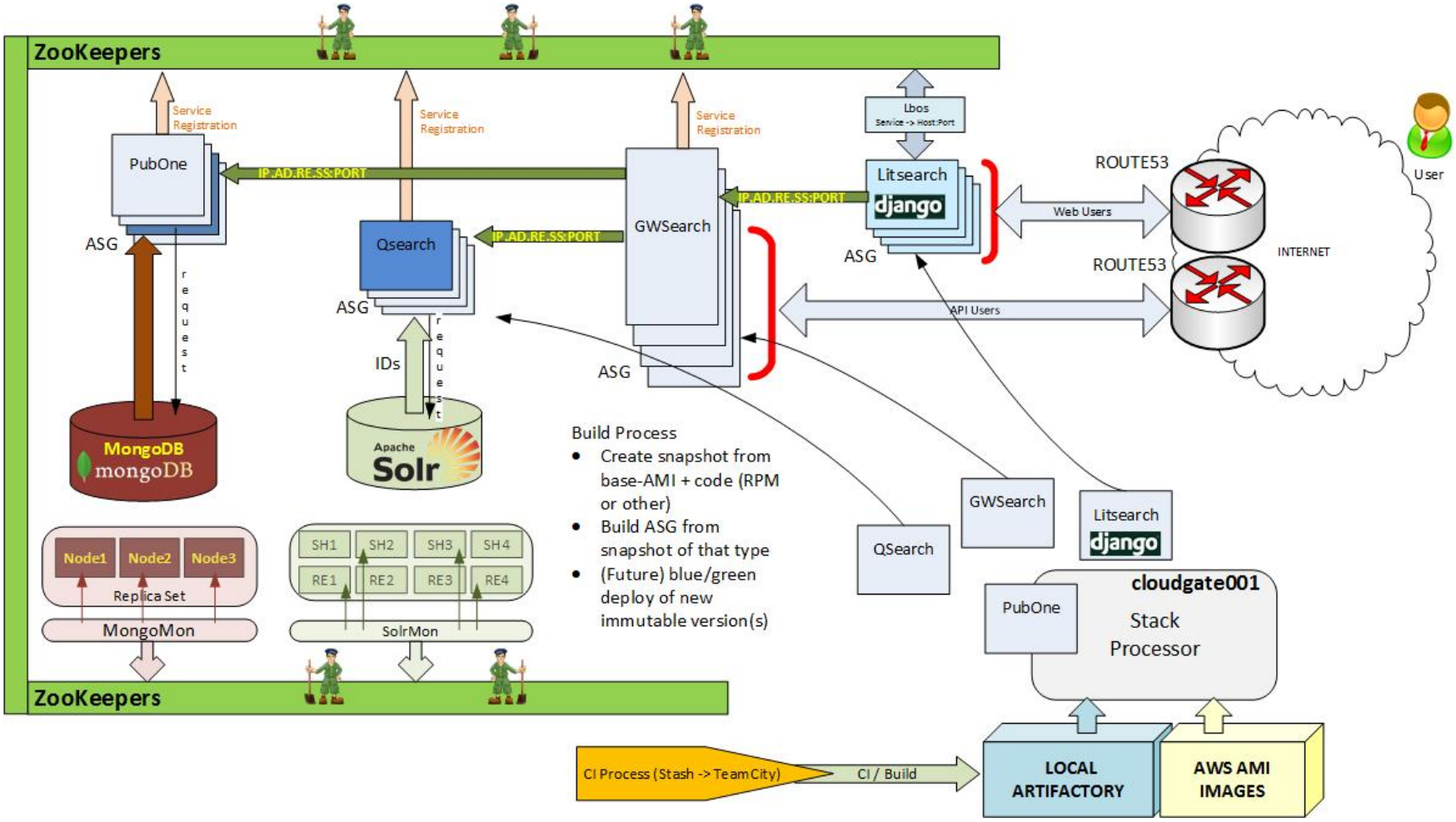


CORRECT



Slave Stack with replication from Master




ZooKeepers
Service Registration
Service Registration
Service Registration
PubOne
IP.AD.RE.SS:PORT
Qsearch
IP.AD.RE.SS:PORT
GWSearch
IP.AD.RE.SS:PORT
Lbos
Service -> Host:Port
Litsearch
django
ASG
ASG
ASG
ASG
request
request
IDs
ROUTE53
Web Users
ROUTE53
API Users
INTERNET
User
MongoDB
mongoDB
Apache
Solr
Build Process
Create snapshot from base-AMI + code (RPM or other)
Build ASG from snapshot of that type
(Future) blue/green deploy of new immutable version(s)
Node1
Node2
Node3
Replica Set
MongoMon
SH1
SH2
SH3
SH4
RE1
RE2
RE3
RE4
SolrMon
QSearch
GWSearch
Litsearch
django
cloudgate001
PubOne
Stack Processor
ZooKeepers
CI Process (Stash -> TeamCity)
CI / Build
LOCAL ARTIFACTORY
AWS AMI IMAGES

Creating an AWS Solr Stack

1. Set up AWS instance
2. Register Solr instance in ZK
3. Upload Solr configuration to ZK
4. Determine shard count
5. Generate replication plan
6. Create collection
7. Apply security settings
8. Start solr_mon
9. Enable replication from Master
10. Enable SolrCloud

Acknowledgements & Contacts

- Feature Location Service
  - Valerie Schneider valerie.schneider@nih.gov
  - Peter Meric peter.meric@nih.gov

  Dataflow: Ray Anderson

  Advisors: Brad Holmes, Terence Murphy

  DBA: Craig Oakley

  Track production: NCBI Eukaryotic Genome Pipeline team

- High-Availability Solr
  - Grisha Starchenko grisha.starchenko@nih.gov