CYWG (CyberInfrastructure Working Group)
October 2014

iDigBio Architecture

Alex Thompson
Advanced Computing and Information Systems Laboratory (ACIS)
University of Florida

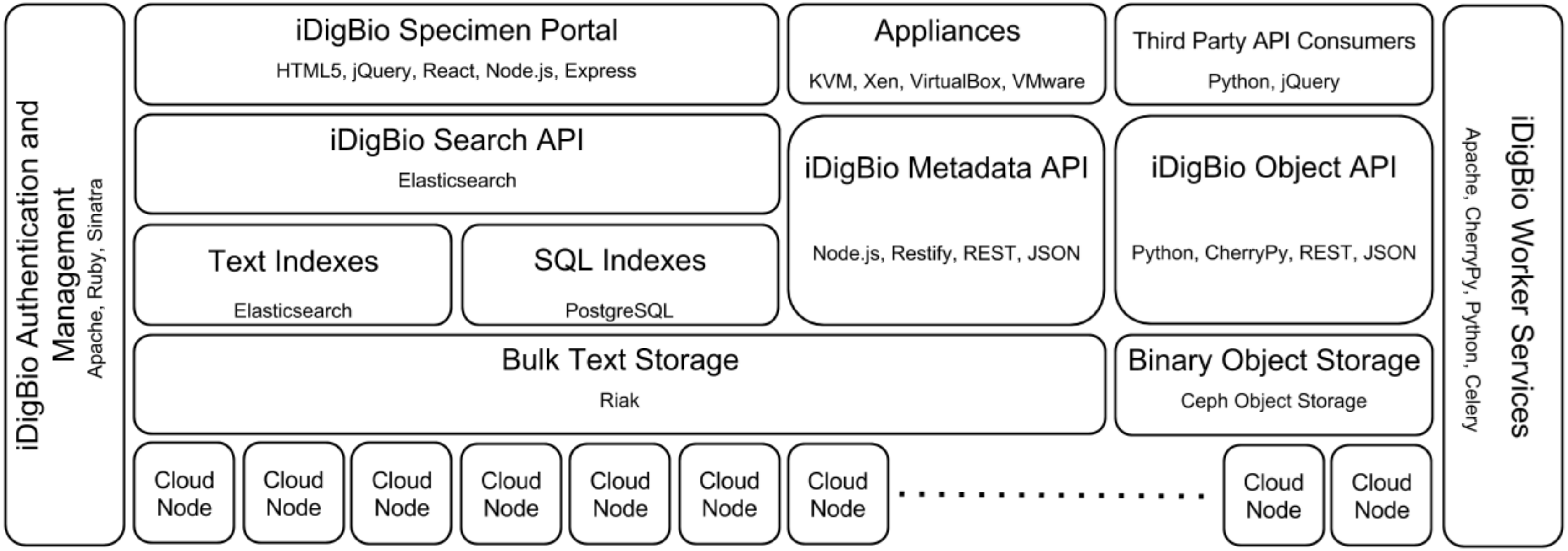
✉ godfoder@acis.ufl.edu

iDigBio is funded by a grant from the National Science Foundation's Advancing Digitization of Biodiversity Collections Program (Cooperative Agreement EF-1115210). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. All images used with permission or are free from copyright.

Open Microservice Architecture

- All iDigBio services should expose a REST API and, if practical, be completely public.
- Each individual service should be as minimal as possible.
- Services should build on top of other services.
- Services should optimize for speed, and scale by supporting many running copies.
- Services should respond quickly, if necessary deferring long running jobs to a centralized background worker system.
- The Portal is the top level service, acting as a consumer and interface for the majority of idigbio services.

Architecture Components



iDigBio Authentication and Management
Apache, Ruby, Sinatra

iDigBio Specimen Portal
HTML5, jQuery, React, Node.js, Express

Appliances
KVM, Xen, VirtualBox, VMware

Third Party API Consumers
Python, jQuery

iDigBio Search API
Elasticsearch

iDigBio Metadata API
Node.js, Restify, REST, JSON

iDigBio Object API
Python, CherryPy, REST, JSON

Text Indexes
Elasticsearch

SQL Indexes
PostgreSQL

Bulk Text Storage
Riak

Binary Object Storage
Ceph Object Storage

Cloud Node | Cloud Node | Cloud Node | Cloud Node | Cloud Node | Cloud Node | Cloud Node | . . . . | Cloud Node | Cloud Node

iDigBio Worker Services
Apache, CherryPy, Python, Celery

Basic Backend Components

Mostly private, the data in these services is very low level and harder to consume.

- Bulk Text Storage - Riak
- Binary Object Storage - Ceph
- Management Data - PostgreSQL

Bulk Text Storage

- Custom data model
- Stores every version of a record ever ingested into iDigBio
- Most of the data is treated as immutable once written
- 50+ Million Objects (~120GB of data)

Binary Object Storage

- S3 API
- Stores images, downloads, datasets, and other content for distribution
- Highly redundant storage, can be scaled across multiple datacenters as iDigBio grows
- ~19 Million Objects (~7 TB of data)

Management Data

- PostgreSQL RDBMS
- Relational data between records
- Fast bulk lookup - Lists, ID Resolution
- Service authentication tokens

Tier 1 Services

Built directly on top of the backends, forms the core of iDigBio's offerings

- Search (Read-only)
- Raw access (Public Read, Write with API Key)
- Object API (Public Read, Write with API Key)

Search

- Powered by ElasticSearch http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/index.html
- (Lightly) Processed versions of the raw records
- Moving towards a model with more heavily processed data, this will be called out explicitly in the data and access to search on raw data will be maintained.
- Documented at: https://www.idigbio.org/wiki/index.php/IDigBio_API_v1_Specification#Search

http://search.idigbio.org/idigbio/records/_search?q=stateprovince:arkansas

```
{
        hits:
        {
                total: 93138,
                max_score: 6.523244,
                hits:
                [
                        {
                                _index: "idigbio-1.4.0",
                                _type: "records",
                                _id: "27b9f1a5-bb52-4da1-8fee-40bb569aaaf2",
                                _score: 6.523244,
                                _source:
                                {
                                        family: "plethodontidae",
                                        recordset: "348f4784-4786-45be-8d0f-85f2b189eba8",
                                        stateprovince: "arkansas",
                                        county: "polk",
                                        phylum: "chordata",
                                        catalognumber: "231195",
                                        specificepithet: "brimleyorum",
                                        continent: "north america",
                                        datemodified: "2014-05-03",
                                        uuid: "27b9f1a5-bb52-4da1-8fee-40bb569aaaf2",
                                        basisofrecord: "preservedspecimen",
                                        collector: "sever and dundee",
                                        institutioncode: "ummz",
                                        verbatimlocality: "united states; arkansas; polk;",
<SNIP>
```

Search

- Supports complex queries with an HTTP Post
- Supports Aggregates for summaries, or to return a lot of data quickly - Geohashing for fast heat maps, Date histograms, term counting and others.
- http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/search-aggregations.html
- Aggregates can be nested to further increase the api’s capabilities - Need to plot geographical changes over time? Nest a geohash_grid inside a date_histogram.

Raw Access

- Minimal REST front end to the core iDigBio data
- Accessible as one huge list, or per recordset.
- Supports retrieval of any version of the record (search is only the most current version)
- Backend processes automatically update the search API when new data is submitted.
- Documented at: https://www.idigbio.org/wiki/index.php/IDigBio_API

http://api.idigbio.org/v1/records/0000012b-9bb8-42f4-ad3b-c958cb22ae45?version=2

```
{
        idigbio:uuid: "0000012b-9bb8-42f4-ad3b-c958cb22ae45",
        idigbio:etag: "9d2209ef58ddfef276e4a06cad57d106942516c1",
        idigbio:dateModified: "2014-04-21T00:36:29.192Z",
        idigbio:version: "2",
        idigbio:createdBy: "872733a2-67a3-4c54-aa76-862735a5f334",
        idigbio:data:
        {
                dwc:startDayOfYear: "233",
                dwc:specificEpithet: "monticola",
                dwc:recordedBy: "P. Acevedo; A. Reilly",
                dwc:locality: "Coral Bay Quarter, Bordeaux Mountain Road.",
                dwc:habitat: "Sunny roadside.",
                dwc:scientificNameAuthorship: "Hitchc.",
                dwc:occurrenceID: "762944",
                dwc:stateProvince: "Saint John",
                dwc:eventDate: "1987-08-21",
                dwc:collectionID: "urn:uuid:a2e32c87-d320-4a01-bafd-a9182ae2e191",
                dwc:country: "U.S. Virgin Islands",
                idigbio:recordId: "urn:uuid:ed400275-09d7-4302-b777-b4e0dcf7f2a3",
                dwc:collectionCode: "Plants",
                dwc:decimalLatitude: "18.348",
                dwc:occurrenceRemarks: "Small tree. 3.0 m. Bark brown, stems smooth; flowers in buds yellow.",
                dwc:rights: "http://creativecommons.org/licenses/by-nc-sa/3.0/",
                dwc:genus: "Eugenia",
                dwc:family: "Myrtaceae",
                dwc:identifiedBy: "Andrew Salywon, Jan 2003",
                dwc:dynamicProperties: "Small tree. 3.0 m. Bark brown, stems smooth; flowers in buds yellow.",
<SNIP>
```

Object API

- Powers the Image Ingestion Appliance and our data ingestion workflow
- Triggers thumbnail generation for images, processing code for datasets via background jobs.
- Can look up objects by ETag (hash), or a user specified file reference.
- Currently no public documentation, but it is dead simple and I can generate documentation if needed.

```
$ curl -X POST -F file=@/home/godfoder/Downloads/4db3b81fddbf08d77ff5c23283e4ac39 -F
filereference="urn:uuid:someguid" "http://api:key@beta-media.idigbio.org/upload/images" | json_pp
{
   "file_reference" : "urn:uuid:someguid",
   "content_type" : "application/octet-stream",
   "file_url" : "http://beta-media.idigbio.org/lookup/images/4db3b81fddbf08d77ff5c23283e4ac39",
   "file_md5" : "4db3b81fddbf08d77ff5c23283e4ac39",
   "object_type" : "images",
   "file_size" : 6708418,
   "file_name" : "4db3b81fddbf08d77ff5c23283e4ac39"
}
```

Download:
http://beta-media.idigbio.org/lookup/images/4db3b81fddbf08d77ff5c23283e4ac39

Pretty much anything that can make an HTTP post request with file contents and authentication can be used as a client. Including very simple HTML forms. The filereference parameter contains the GUID. An API UUID and Key are required before it will work though.

Tier 2+ Services

Built directly on top of other services, offering specialized functionality.

- Download
- Coming soon: GBIF Upload
- Portal - Covered in a different presentation

Download Service

- No size limitation (can download all of iDigBio).
- Takes in a specialized formulation of a search query (designed to be fairly easy to compose).
- Query format supports everything that can be done in the portal right now.
- Uses background tasks and optimized elasticsearch queries to build a darwin core archive for consumers.
- Easiest way to get bulk data out of iDigBio.

```
$ curl -X POST -F 'query={"country":"morocco"}' -F 'email=godfoder@acis.ufl.edu' http://csv.idigbio.org/ | json_pp
{
   "query_hash" : "fe46090ca730e797a76391f1812f3dfdcda1a2f1",
   "complete" : false,
   "task_status" : "PENDING",
   "status_url" : "http://csv.idigbio.org/status/ee3c9bdf-e239-4328-898f-daea21991593",
   "query" : {
      "country" : "morocco"
   }
}
```

Returns a status_url which can be polled until the download is complete (complete: true), at which point a download_url will be added to the response.

```
curl http://csv.idigbio.org/status/ee3c9bdf-e239-4328-898f-daea21991593 | json_pp
{
   "download_url" : "http://s.idigbio.org/idigbio-downloads/ee3c9bdf-e239-4328-898f-daea21991593.zip",
   "complete" : true,
   "task_status" : "SUCCESS",
   "query_hash" : "fe46090ca730e797a76391f1812f3dfdcda1a2f1",
   "query" : {
      "country" : "morocco"
   },
   "status_url" : "http://csv.idigbio.org/status/ee3c9bdf-e239-4328-898f-daea21991593"
}
```

GBIF Upload

- Built on top of the download system
- Automatically builds a DwC-A for every dataset
- Will track associations with GBIF Publishers/Datasets and provide any missing data to GBIF.
- A work in progress, will start to move in to public trails soon.
- The end goal is to have 100% of iDigBio data accessible via GBIF one way or another.

Questions? Comments?

Want to use our services and APIs but need help? E-mail idigbio@acis.ufl.edu to reach the entire ACIS team at once.

www.idigbio.org

facebook.com/iDigBio

twitter.com/iDigBio

vimeo.com/idigbio

idigbio.org/rss-feed.xml

webcal://www.idigbio.org/events-calendar/export.ics

iDigBio is funded by a grant from the National Science Foundation's Advancing Digitization of Biodiversity Collections Program (Cooperative Agreement EF-1115210). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. All images used with permission or are free from copyright.