An API To Feed the World

The overarching goal of iPlant’s CI is to help biologists focus on conducting science to answer biological research questions, rather than dealing with computing resource details, learning new analysis software with each new question, or converting data between file types. The iPlant Foundational API provides the core functionality to make this possible. Through the Foundational API users can create jobs, data, workflows, and relationships, connect with their science in the ways they work best, and collaborate with other people inside and outside of iPlant. In this talk I will briefly describe the anatomy of the Foundational API, illustrate how it supports the opt-in design of the iPlant CI, and provide context on how this particular API can help feed the world.
An API To Feed the World

Rion Dooley
Everyone Communicates, Few Connect

- Many problems: using two grand challenge problems (iPG2P, iPToL) as models for developing a cyberinfrastructure that will be broadly applicable to the community as a whole.
- Many workers:
  - 4000-6000 active people in the plant science community
  - > 100 groups actively developing tools for the community
- Different skill sets:
  - Developers (Perl, Python, bash, Matlab, .NET, C, Fortran, Java, C++, R, MPI,...)
  - Bio hackers (a bit of this, just a bit more of that)
  - Consumers (point and click desktop/web interface)
Everyone Communicates, Few Connect

Lots of different ways that computational biology is being conducted

<table>
<thead>
<tr>
<th>Command line scripts</th>
<th>Advanced workflows</th>
<th>Data rich</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom serial and parallel codes</td>
<td>Commercial codes</td>
<td>High throughput</td>
</tr>
<tr>
<td>Ensemble runs across systems</td>
<td>Long running/short running</td>
<td>Distributed data</td>
</tr>
<tr>
<td>Advanced visualization</td>
<td>Parametric sweeps</td>
<td>Database driven</td>
</tr>
</tbody>
</table>
Everyone Communicates, Few Connect

iPlant architecture does its best to accommodate all these approaches to conducting science through a “grocery store approach.”

“Take what you need and leave the rest for later.”

The Foundational API is the point of entry for a broad suite of tools and a value added feature of the CI for command line lovers.
Everyone Communicates,
Few Connect
Foundational API
https://foundation.iplantcollaborative.org

- REST API
- HTTP Basic and Token-based authentication (true OAuth coming soon)
- Synchronous and asynchronous
- Modeled after several popular social and industry APIs (Yelp!, Dropbox, PayPal, etc.)
- Unhealthy obsession with pretty URLs
Foundational API
https://foundation.iplantcollaborative.org

General design concepts

• Should know that it doesn’t know everything.
• All services should add value to the underlying middleware, not bloat it.
• Should make sharing and showing off easier.
• Should allow consumers to do the right thing.
Foundational API
https://foundation.iplantcollaborative.org

Current Services
- Auth
- Profile
- IO
- Data
- Apps
- Jobs

Future Services
- Event
- Mashup (workflow)
Foundational API
https://foundation.iplantcollaborative.org/auth-v1/

Auth Service

/auth-v1/
GET
Validates the username and password/token used to authenticate.

/auth-v1/
PUT
Renews the token used to authenticate for 2 hours.

/auth-v1/
DELETE
Expires the token used to authenticate.
Foundational API

https://foundation.iplantcollaborative.org/auth-v1/

Auth Service

/auth-v1/

POST
Issues a token for a user. The tokens issued can be used interchangeably by the auth service and the rest of the Foundation API services just like the actual user passwords.

/auth-v1/list

GET
Returns JSON array with all active tokens for the authenticated user.
Foundational API

https://foundation.iplantcollaborative.org/profile-v1/

Profile Service

/profile-v1/profile/
GET
Returns JSON description of the authenticated user profile

/profile-v1/profile/username/<username>
GET
Returns JSON description of the user with the given username

/profile-v1/profile/search/username/<username>
GET
Returns JSON array containing description of users with matching usernames
Foundational API

https://foundation.iplantcollaborative.org/profile-v1/

Profile Service

/get/profile-v1/profile/search/name/<name>

GET

Returns JSON array containing descriptions of the users matching name

/get/profile-v1/profile/search/email/<email>

GET

Returns JSON array containing description of users with matching email addresses.
Foundational API
https://foundation.iplantcollaborative.org/io-v1/

IO Service

/io-v1/io/list/<username>/<path>
GET
Returns JSON array containing description of the file/folder referenced by <path>

/io-v1/io/<username>/<path>
GET, POST, DELETE, PUT
Performs operations on files and folders

/io-v1/io/share/<username>/<path>
GET, POST
Performs operations on file and folder share permissions
Foundational API
https://foundation.iplantcollaborative.org/io-v1/

Data Service

/io-v1/data/transforms
GET
Returns JSON list of all available transforms

/io-v1/data/transforms/<username>/<path>
GET
Returns JSON list of available transforms for a specific file

/io-v1/data/transform/<transform_name>
GET
Returns JSON description of the transform
Foundational API

https://foundation.iplantcollaborative.org/io-v1/

**Data Service**

```
/io-v1/data/tag/<tag_name>
```

**GET**
Returns JSON list of transforms tagged with the given term

```
/io-v1/data/async/transform/<transform_name>/
<username>/<path>
```

**POST**
Returns the file transformed from its original format to the named format and staged to a location defined by the user.

```
/io-v1/data-sync/transform/<transform_name>/
<username>/<path>
```

**GET**
Returns the file transformed into the named format.
Foundational API
https://foundation.iplantcollaborative.org/apps-v1/

Apps Service

/apps-v1/apps
POST
Add or update an app you own by posting an app description file

/apps-v1/apps/list
GET
Returns JSON array containing a list of all public apps

/apps-v1/apps/share/list
GET
Returns JSON array containing a list of all your public and shared apps
### Foundational API

**https://foundation.iplantcollaborative.org/apps-v1/**

**Apps Service**

<table>
<thead>
<tr>
<th>Route</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/apps-v1/apps/name/&lt;name&gt;</code></td>
<td>GET Returns a list of apps with name matching the search term</td>
</tr>
<tr>
<td><code>/apps-v1/apps/tag/&lt;tag&gt;</code></td>
<td>GET Returns a list of apps with tag matching the search term</td>
</tr>
<tr>
<td><code>/apps-v1/apps/term/&lt;term&gt;</code></td>
<td>GET Returns a list of apps with ontology matching the search term</td>
</tr>
</tbody>
</table>
# Foundational API

https://foundation.iplantcollaborative.org/apps-v1/

## Job Service

<table>
<thead>
<tr>
<th>Path</th>
<th>Method(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/apps-v1/job</td>
<td>POST</td>
<td>Submit a job request</td>
</tr>
<tr>
<td>/apps-v1/job/&lt;job_id&gt;</td>
<td>GET, POST, DELETE</td>
<td>Query for information, resubmit, or delete a job</td>
</tr>
<tr>
<td>/apps-v1/io/share/&lt;username&gt;/&lt;path&gt;</td>
<td>GET, POST</td>
<td>Performs operations on job share permissions</td>
</tr>
</tbody>
</table>
Foundational API
https://foundation.iplantcollaborative.org/apps-v1/

Job Service

GET
//apps-v1/job/<job_id>/input
Returns JSON array containing descriptions of the inputs for the job

GET
//apps-v1/job/<job_id>/output/list/<path>
Returns JSON array of job output files (similar to IO, but unique to the job)

GET, POST
//apps-v1/job/<job_id>/output/<path>
Downloads the file at the given path relative to the job’s output folder
Coming sooner...

Event Service

/event-v1/
GET, POST
Register a new event or get a snapshot of all your registered events

/event-v1/<event_id>
GET, POST, DELETE
Query, update, and delete event subscriptions
Coming later...

Mashup Service

/mashup-v1/
POST
Submit a mashup for execution

/mashup-v1/<mashup_id>
GET, POST, DELETE
Returns JSON descriptions of a specific mashup

/mashup-v1/list
GET
Returns a list of JSON descriptions of your mashups
Coming later...

**Mashup Service**

/mashup-v1/tags/<tag>
GET
Search for mashups by tag

/mashup-v1/share/<mashup_id>
GET, POST, DELETE
Manage share permissions on a specific mashup

/mashup-v1/share/<mashup_id>
POST, DELETE
Performs operations on job share permissions
Demo App
https://foundation.iplantcollaborative.org/iplant-test
Pros & Cons

Pros

+ Very web and web service friendly.
+ Very scalable.
+ Runs anywhere.
+ Take what you need, leave the rest (for later).
Pros & Cons

Cons

- Always a struggle finding friendliest URLs.
- Underlying systems are still the same, so some endpoints (those interacting directly with the systems) are not as fast as pure database driven services.
- IO and Data are really good for async fire and forget data movement, not so good for direct BIG DATA upload. Use other iPlant Data Storage tools.
Conclusion

• Lots of workers, dearth of tools
• Everyone communicates, few connect
• We built it, now they're coming: iPlant Discovery Environment, Cypress, PhytoBisque, CSIRO, IBP
• This means: Food in Africa, a brighter future, and Dan's favorite…
Cheeseburgers