

Data Communication Plan

Expectations

The data communication structure outlined in this document intends to be ideal for meeting the following expectations:

- **Redundant** - Can fall back to alternate access methods in case of unavailability
- **Offline** - Caches data so it is available while offline; updates the data when online
- **Proxied** - Machines with limited network access can access the data from other machines that either have direct access or cached copies, if proxying is set up ahead of time.

Technologies

- HTTP API client
- Configurable fallback system
- Configurable authentication
- Local cache database
- Shim API to proxy other machines, with authentication

Structure

The core of this project's data access flow is the concept of a **source**, which is an abstraction over the API's functionality. There are multiple concrete implementations of sources:

- **HTTP API**: Source that directly uses an HTTP client to access the API, with authentication
- **Fallback Multiplexer**: Source that can be configured with multiple subsources; selects the first working subsourse
- **Cache** - Accesses data from a local database

For proxying, there is a reimplementatation of the API that serves from a configured source, allowing other machines to access the data by setting the proxy as an endpoint. The reimplemented API has its own configurable authentication system with granular control.