Location Services R Packages

This location services R package will be a meta-package. A meta-package is a package that is comprised solely of other packages. The meta-package will be called `{arcgis}`.

> The goal of {arcgis} is to make ArcGIS Platform services available to R users.

Context: ArcGIS Platform and API for Python

ArcGIS Platform Services are:

- Basemaps
- Geocoding and Search
- Routing and Directions
- Data Hosting
- Data Visualization
- Maps and Data
- Spatial Analytics
- GeoEnrichment
- Places

The ArcGIS API for Python provides access to location services. The ArcGIS API for Python acts as a good reference implementation. The key features of the API for Python are:

- Authentication
- Mapping
- Geocoding
- Routing
- Spatial analysis
- Data enrichment
- Deep learning
- Administration
- Content management
- Geoprocessing

Initial R package Goals:

An initial R package targets only "authentication", "data hosting", and "content management." We want to provide R users with an Esri developed way to interact with hosted data.

Authentication

The API for Python lists 5 ways to authenticate with a service. These are:

- Anonymously (no auth required)
- Using built-in accounts
- Enterprise identity stores
- Using OAuth2
- Using ArcGIS Pro

An initial release will focus on OAuth2, ArcGIS Pro, and username and password. This will provide the vast majority of R users with a way authorize to location services.

Data Hosting & Content Management

Location services and the API for Python provide very thorough functionality for managing remote content. For an initial release, this R package is focused on covering the simple use cases. These are:

- reading from and writing to a Feature Service,
- updating and deleting records from a Feature Service,
- and reading from an Image Service.

More advanced content management such as search, creating groups, users, and modifying item properties is out of scope for an initial release.

Location Services R package structure

{arcgis} will be a meta-package. A meta-package is a package that only imports other packages.

The meta-package will consist of `{arcgisutils}` and `{arcgislayers}` initially. Packages for geocoding and routing can be developed then included in the meta-package further down the road.

A meta-package approach enables more modular development, lowers the barrier for community contributions, and reduces the number of needed dependencies for production code that uses our package.

{arcgisutils}

arcgisutils is the foundational R package from which everything is built upon. It is targeted at developers who want to build low-level functions or packages on location services.

arcgisutils contains functions for:

- creating Esri JSON
- parsing Esri JSON
- R to Esri type conversion
- authorization
- handling requests

{arcgislayers}

arcgislayers is built upon arcgisutils' helper functions for creating and parsing Esri JSON. It provides functions for:

- reading remote data metadata
- reading remote data into R formats
- adding, deleting, and updating features in feature services
- adding and publishing R objects as feature services

{arcgis}

arcgis loads all lower-level R packages. It is akin to loading the `{tidyverse}` but much much smaller and less invasive. This is the current behavior:

```
library(arcgis)
```

```
Attaching core arcgis packages:
  - {arcgisutils} v0.1.0
  - {arcgislayers} v0.1.0
```