

What is an API?

Application programming interface (API) is an interface in computer science that defines the ways by which an application program may request services from libraries and/or operating systems. An API determines the vocabulary and calling conventions the programmer should employ to use the services. It may include specifications for routines, data structures, object classes and protocols used to communicate between the requesting software and the library.

http://en.wikipedia.org/wiki/Application_programming_interface

Why is this important?

Starting with R5.1 RUcore will begin to employ API's to perform some functionality. The API must be placed in a web accessible directory that cannot change to ensure future interoperability with agents that might use the service. The local RUcore development team might not always be in control of the agents using the API, thus moving the API location would be damaging to those agents that cannot be updated with the new location information.

Proposal

We create an API directory under the rucore.libraries URL .

`http://rucore.libraries.rutgers.edu/api/<API 1>/`

...

`http://rucore.libraries.rutgers.edu/api/<API N>/`

Example using the an API for depositing materials might be

`http://rucore.libraries.rutgers.edu/api/deposit/`

What should an API directory structure look like?

An API is essentially made up of services (methods) it provides. Each method an API employs should be separated logically from each other. Also, API's must provide documentation stating how it is to be used.

Proposal

`/<API NAME>/<METHOD 1>/`

...

`/<API NAME>/<METHOD N>/`

`/<API NAME>/doc/`

Example using for depositing materials might be

`http://rucore.libraries.rutgers.edu/api/deposit/add/` (*deposits an item*)

`http://rucore.libraries.rutgers.edu/api/deposit/append/` (*appends information on an already deposited item*)

`http://rucore.libraries.rutgers.edu/api/deposit/remove/` (*removes a deposited item*)