REST API proposal

Proposal for making ACF API fully RESTful

Making the API fully RESTful involves three changes:

- Using HTTP verbs in an appropriate manner
- Defining a URL for every object
- Not passing arguments other than a path to GET methods

This is the proposal for mapping the current functionality to RESTful functionality.

The actual current commands are as follows. Note that NO changes are proposed for the output objects of the commands at this time. Also, note that all connection_name fields in URLs are encoded such that "/" characters do not appear in them; specifically, the sequences ".." and ".+" represent "." and "/", respectively.

Current command	Current argument format	Proposed verb	Proposed command
outputconnector/list	N/A	GET	outputconnectors
authorityconnector/list	N/A	GET	authorityconnectors
repositoryconnector/list	N/A	GET	repositoryconnectors
outputconnection/list	N/A	GET	outputconnections
outputconnection/get	{"connection_name":*<connection_name>*}	GET	outputconnections/*<output_connection_name>*
outputconnection/save	{"outputconnection":*<output_connection_object>*}	PUT	outputconnections/*<output_connection_name>* [note: the connection name is optional, since the object contains the name as well]
outputconnection/delete	{"connection_name":*<connection_name>*}	DELETE	outputconnections/*<output_connection_name>*
outputconnection/checkstatus	{"connection_name":*<connection_name>*}	GET	status/outputconnections/*<output_connection_name>*
outputconnection/execute/*<command>*	{"connection_name":*<connection_name>*, *<argument_data>*}	GET	info/outputconnections/*<output_connection_name>*/*<command>*
authorityconnection/list	N/A	GET	authorityconnections
authorityconnection/get	{"connection_name":*<connection_name>*}	GET	authorityconnections/*<authority_connection_name>*
authorityconnection/save	{"authorityconnection":*<authority_connection_object>*}	PUT	authorityconnections/*<authority_connection_name>*
authorityconnection/delete	{"connection_name":*<connection_name>*}	DELETE	authorityconnections/*<authority_connection_name>*
authorityconnection /checkstatus	{"connection_name":*<connection_name>*}	GET	status/authorityconnections/*<authority_connection_name>*
repositoryconnection/list	N/A	GET	repositoryconnections
repositoryconnection/get	{"connection_name":*<connection_name>*}	GET	repositoryconnections/*<repository_connection_name>*
repositoryconnection/save	{"repositoryconnection":*<repository_connection_object>*}	PUT	repositoryconnection/*<repository_connection_name>*
repositoryconnection/delete	{"connection_name":*<connection_name>*}	DELETE	repositoryconnections/*<repository_connection_name>*
repositoryconnection /checkstatus	{"connection_name":*<connection_name>*}	GET	status/repositoryconnections/*<repository_connection_name>*
repositoryconnection/execute/*<command>*	{"connection_name":*<connection_name>*, *<argument_data>*}	GET	info/repositoryconnections/*<repository_connection_name>*/*<command>*
job/list	N/A	GET	jobs
job/get	{"job_id":*<job_identifier>*}	GET	jobs/*<job_identifier>*
job/save	{"job":*<job_object>*}	PUT	jobs/*<job_identifier>* [Save existing]
job/save	{"job":*<job_object>*}	POST	jobs [Create new, returning job identifier]
job/delete	{"job_id":*<job_identifier>*}	DELETE	jobs/*<job_identifier>*
jobstatus/list	N/A	GET	jobstatuses
jobstatus/get	{"job_id":*<job_identifier>*}	GET	jobstatuses/*<job_identifier>*
jobstatus/start	{"job_id":*<job_identifier>*}	PUT	start/*<job_identifier>*
jobstatus/abort	{"job_id":*<job_identifier>*}	PUT	abort/*<job_identifier>*
jobstatus/restart	{"job_id":*<job_identifier>*}	PUT	restart/*<job_identifier>*
jobstatus/pause	{"job_id":*<job_identifier>*}	PUT	pause/*<job_identifier>*
jobstatus/resume	{"job_id":*<job_identifier>*}	PUT	resume/*<job_identifier>*