Item 319

git_comments:

1. Should not set trusted=true in configSet

git_commits:

1. **summary:** SOLR-14907: Adding V2 API for ConfigSet Upload. (#1996)
   message: SOLR-14907: Adding V2 API for ConfigSet Upload. (#1996)

github_issues:

github_issues_comments:

github_pulls:

1. **title:** SOLR-14907: Adding V2 API for ConfigSet Upload.
   body: https://issues.apache.org/jira/browse/SOLR-14907 Adding v2 APIs.

github_pulls_comments:

1. Are there any concerns now that the API has been sorted?
2. LGTM

github_pulls_reviews:

1. Let's use a proper REST mechanism here `path ="/cluster/configs/{configname}/*`
2. same as above
3. Would you be ok with `/cluster/configs/{configname}/files/*`? That's very restful, since it would be updating the `*` `file` under the `{configname}` `config`
4. I've updated the PR to reflect this new API.
5. I'm Ok with both `/cluster/configs/{configname}/files/*` and `/cluster/configs/{configname}/*` @janhoy @dsmiley @sigram WDYT?
6. I don't see the point of the "/files/" component of the path. The files within the configSet are directly in it, so I think it should be that way. I think of this very much like a file in a folder, where the folder is the configSet. Just checking that we support nested structure within the configSet?
7. We do support the nested structure. The `/files/` part defines what the * after it is, and allows us to extend in the future. For example possible adding a `/cluster/configs/{configname}/properties` PUT action to update the properties for that config. It fits pretty well in the [guidelines](https://restfulapi.net/resource-naming/). It's not that important to me though if y'all agree with taking it out.
8. >Just checking that we support nested structure within the configSet? We could. But, I'm OK to add support for that later
9. To support properties manipulation in the future, I think that could be done via a POST to the particular path to the properties file in the configSet (I forget if it exists already or maybe should be exactly params.json) with commands in the JSON to delete/set/update etc.
10. Ok sounds good, that would certainly work. I'll take out the `/files/` part of the path, and update the PR and Jira.
11. Can we just use a boolean here?
12. I'm not a REST specialist but I think I would prefer to leave the `files` component of the path in. Wouldn't this allow changing other things in the future and make the API more self describing? Would it really be easy to add it later? If we wanted to add properties for example, how to tell if `/cluster/configs/{configname}/properties/whatever` is a property `whatever` or if `properties/whatever` is actually a file?
13. Also "file" is a pretty generic abstraction that can represent many things (a schema, properties, synonyms etc). We might eventually want to address such ressources by what they actually are (schema, properties, synonyms) rather than by the way in which they are stored (files).
14. I agree @murblanc, it lets us future-proof the API even if we don't have current plans on expanding beyond files.

15. All very debatable of course. Removing "/files/" doesn't prevent expanding beyond files, it's just that, for example, if we add a POST to say /configsetname/properties then we can't have a file named simply "properties". I will shed no tear over this loss :-). I like the simplicity of not having the needless "/files/".
16. @dsmiley, can users of Solr pick random names for some ConfigSet files? I believe code will be easier to understand/maintain if we don't have to parse file names to detect if they are files or some other form of separator (and then the API would be inconsistent: files will not have its `/files` segment in the path whereas properties will be defined after `/properties`). Examples in https://restfulapi.net/resource-naming/ would be a vote for keeping `/files` in the path IMO.
17. People can add basically whatever named files they want and reference them from the primary config files (e.g. solrconfig.xml). This discussion is similar to field naming in our schema. Some names are special, like `_version_`. If you try to have your own field named as such, then you're asking for trouble. Likewise, we might suggest that users avoid file names with a leading underscore. Simple enough and keeps the "/files" verbosity out. Also there is a difference in HTTP verbs... in this issue here we're using PUT to put a file. But for an interactive editing of config elements, that would use POST.

jira_issues:

jira_issues_comments: