**BCF 2.0 RESTful API**

All API access is over HTTPS. Data is sent as query parameters and received as JSON.

* Every resource has a corresponding Json Schema.
* Json Hyper Schema for link definition
* Authentication OAuth2

**Paging, Sorting**

* [Link header introduced by RFC 5988](http://tools.ietf.org/html/rfc5988#page-6).
* custom HTTP header ***X-Total-Count***

eg: /v1.0/projects?page=1&pageSize=5&sortColumn=type&sortDirection=desc

page : requested page (it could be record, item, itemNumber, pageNumber, section, sectionNumber, index, offset)

pageSize : requested items per page (records, recordsPerPage, itemSize, items, sections, range)

sortColumn : sort by column name (sortBy, orderBy, sortCol)

sortDirection : sorting direction ( ascending is default or, desc=true/false, sort, sortDir)

As for paging/sort, instead of sortDirection we could use a minus-sign for descending or no sign if ascending. Also, I think that pageSize should be page\_size since our json keys are on that format (e.g. verbal\_status).   
  
For example:   
/comments?sort=-modified\_date&page=1&page\_size=10

**Filtering**

|  |  |
| --- | --- |
| **Operator** | **Operation** |
| | | Greater than |
| < | Less than |
| \* | Wild card for contains (eg: es should yield you "less", "guess" etc) |
| = | Equal to |
| |= | Greater than or equal to |
| <= | Less than or equal to |
| != | Is not equal to |

eg: v2/<resouce\_name>?filter=<json\_attribute><operator><value><Escape codes of AND or OR><json\_attribute><operator><value>......

v2/messages?filter=unread=true%26receiver.id=guid%26date>12.12.2013

Escape codes: <http://www.december.com/html/spec/esccodes.html>

Regarding the filter, do you (or should we) allow for nested filters? E.g filter=(key1=value1 | key2=value2) & key3!=value3 (URL-encoded).

Thanks for your answer. Regarding Rune's question

1.

E.g filter=(key1=value1 | key2=value2) & key3!=value3 (URL-encoded). --> As I already told you we use the conventional "&" to combine different query parameters (in our case operations)

eg: v2/projects?page=1&pageSize=5&sortColumn=type&filter

But, filter has a special criteria because we could combine different properties into one. eg: filter=unread=true%26receiver.id=guid%26date>12.12.2013. We investigated a lot and initially implemented

; --> for AND

| --> for OR

as the delimiter

that is--> filter=unread=true;receiver.id=guid;date>12.12.2013

because "&" was technically not possible as we already use "&" to differentiate different operations(filter, sort etc). But, our client side guru Alexey gave a better idea which will always work. That is to use the escape characters (%26 for "&") which is the correct one in my opinion. So, in that case it is very easy implementation both for the client/server side and it will always work.

2. Regarding using symbols in url (minus sign & page-size)

I think we should follow the query string convention for using the query strings. I think the convention like "verbal\_status" we should always follow it for json property but not in the url. I have had problems using the symbols (- or \_) in the query strings. I can't exactly remember the programming language (maybe php) and our javascript expert also confirms that he has had that problem. Also, using "-" for Desc I think is not self explainable.

But, do you generally agree with the naming ?

page : requested page (it could be record, item, itemNumber, pageNumber, section, sectionNumber, index, offset) --> maybe pageNum is more self explainable ?

pageSize : requested items per page (records, recordsPerPage, itemSize, items, sections, range)

sortColumn : sort by column name (sortBy, orderBy, sortCol) --> maybe sortCol is the right one ?

sortDirection : sorting direction ( ascending is default or, desc=true/false, sort, sortDir) --> maybe sortDir is the right one?

**1**. Yes, I understand and I agree that this is a nice solution.  
  
However, should we in BCF-API allow for nested filters such as the one I showed:   
(key1=value1%7Ckey2=value2)%26key3!=value3   
  
How advanced search should we support?   
  
**2**. Regards to the pageSize vs page\_size. I believe that the common norm is to use \_ (or -) to separate words in URLs. Here are some examples:  
  
<https://developers.facebook.com/docs/graph-api/using-graph-api>:  
- see e.g topic\_filter  
  
<https://dev.twitter.com/docs/api/1.1>:  
- see e.g. screen\_name  
  
<http://developer.netflix.com/docs/REST_API_Reference#jcs-2>:  
- see e.g. max\_result  
  
If what you say is correct, then the clients that do not support \_ or - cannot consume any of the services above.  
  
  
In regards to sortDirection vs minus-prefix, this was actually a suggestion taken directly from the two best practices documents that we shared last week:  
  
<https://drive.google.com/file/d/0B8Wgpx02U7oLcW9qeFVOVGQyNDg/edit?usp=sharing>  
<http://www.vinaysahni.com/best-practices-for-a-pragmatic-restful-api#advanced-queries>  
  
This was just to make the URLs shorter, but I do not have a strong opinion here.   
  
As for the naming, I agree with page, page size and sort direction. Perhaps "sort column" should be "sort by" as a json response does not represent columns per se.

**Caching**

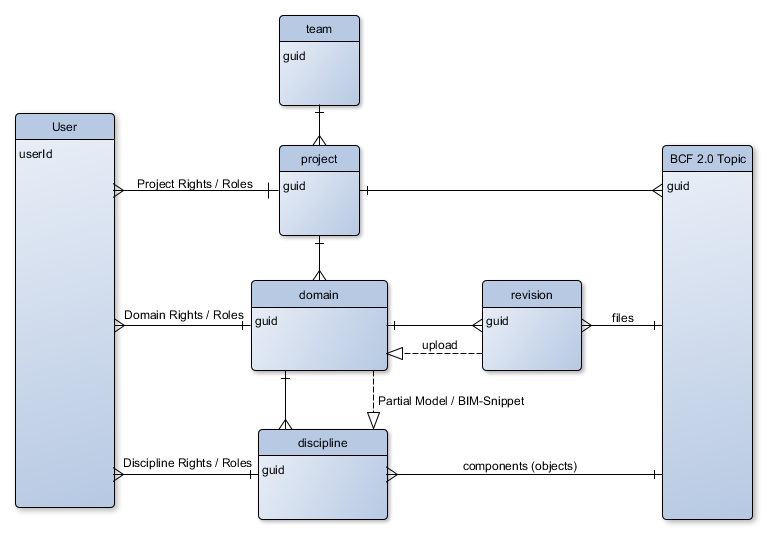
ETAG

**JsonP**

………….

**Structure**

……



**Resources**

……………

|  |  |
| --- | --- |
| ***Version*** | |
| /V0.99 | GET |

|  |  |
| --- | --- |
| ***(Version) Schemas*** | |
| /V0.99/schemas | GET Link Schema |
| /V0.99/schemas/{rel} | GET |

|  |  |
| --- | --- |
| ***Teams*** | |
| /V0.99/teams | GET, POST |
| /V0.99/teams/{guid} | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***Projects*** | |
| /V0.99/ teams/{ guid }/projects | GET, POST |
| /V0.99/ teams/{ guid }/projects/{guid}  /V0.99/projects/{guid} | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***(Projects) Thumbnail*** | |
| /V0.99/ teams/{ guid }/projects/{guid}/thumbnail  /V0.99/projects/{guid}/thumbnail | GET, POST, DELETE |

|  |  |
| --- | --- |
| ***(Projects) Schema*** | |
| /V0.99/ teams/{ guid}/projects /{guid}/schema  /V0.99/ projects /{guid}/schema | GET,POST, PUT, DELETE (extensions.json) |

|  |  |
| --- | --- |
| ***Domains*** | |
| /V0.99/ teams/{ guid }/projects/{guid}/domains | GET, POST |
| /V0.99/ teams/{ guid }/projects/{guid}/domains/{guid}  /V0.99/ domains/{guid} | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***Disciplines*** | |
| /V0.99/ teams/{ guid }/projects/{guid}/ domains/{guid}/  diciplines | GET, POST |
| /V0.99/ teams/{ guid }/projects/{guid}/ domains/{guid}/  disciplines/{guid}  /V0.99/ disciplines/{guid} | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***Revisions*** | |
| /V0.99/ teams/{ guid }/projects/{guid}/divisions/{guid}/revisions | GET, POST |
| /V0.99/ teams/{ guid }/projects/{guid}/divisions/{guid}/revisions/{guid}  /V0.99/revisions/{guid} | GET, DELETE |

|  |  |
| --- | --- |
| ***Topics*** | |
| /V0.99/teams/{ guid }/projects/{guid}/topics | GET, POST |
| /V0.99/teams/{ guid }/projects/{guid}/topics/{guid}  /V0.99/topics/ {guid } | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***(Topics) Revisions*** | |
| /V0.99/teams/{ guid }/projects/{guid}/topics/{guid}/revisions  /V0.99/topics/ {guid }/revisions | GET, POST |

|  |  |
| --- | --- |
| ***(Topics) Divisions*** | |
| /V0.99/teams/{ guid }/projects/{guid}/topics/{guid}/devisions  /V0.99/topics/ {guid }/devisions | GET, POST |

??? is Disciplines the same as Components

|  |  |
| --- | --- |
| ***(Topics) Disciplines*** | |
| /V0.99/teams/{ guid }/projects/{guid}/topics/{guid}/disciplines  /V0.99/topics/ {guid }/disciplines | GET, POST |

|  |  |
| --- | --- |
| ***(Topics) Document References*** | |
| /V0.99/ teams/{ guid }/projects/{guid}/topics/{guid}/  document\_references  /V0.99/topics/ {guid }/ document\_references | GET, POST |
| /V0.99/teams/{ guid }/projects/{guid}/topics/{guid}/  document\_references/{guid}  /V0.99/ document\_references/{guid} | GET, DELETE |

|  |  |
| --- | --- |
| ***(Topic) Related Topics*** | |
| /V0.99/ teams/{ guid }/projects/{guid}/topics/{guid}/related\_topics  /V0.99/topics/{guid}/related\_topics | GET, POST |
| /V0.99/ teams/{ guid }/projects/{guid}/topics/{guid}/related\_topics/{guid}  /V0.99/ topics/{guid}/related\_topics/{guid} | GET, DELETE |

|  |  |
| --- | --- |
| ***(Topics) Comments*** | |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/comments  /V0.99/topics/{guid}/comments | GET, POST |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/comments/{guid}  /V0.99/comments/{guid} | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***(Topics)( Comments) Viewpoints*** | |
| /V0.99/ teams/{id}/projects/{id}/topics/{guid}/comments/{guid}/  Viewpoints  /V0.99/ comments/{guid}/viewpoints | GET |

|  |  |
| --- | --- |
| ***(Topics) Viewpoints*** | |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/viewpoints  /V0.99/topics/{guid}/viewpoints | GET, POST |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}  /V0.99/viewpoints /{guid} | GET, PUT, DELETE |

|  |  |
| --- | --- |
| ***(Topics) (Viewpoints) Bitmap*** | |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}/  bitmap  /V0.99/viewpoints /{guid}/bitmap | GET, POST, DELETE |

|  |  |
| --- | --- |
| ***(Topics) (Viewpoints) Snapshot*** | |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}/  snapshot  /V0.99/viewpoints /{guid}/snapshot | GET, POST, DELETE |

|  |  |
| --- | --- |
| ***(Topics) (Viewpoints) Comments*** | |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}/comments  /V0.99/viewpoints /{guid}/comments | GET, POST |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}/comments/{guid}  /V0.99/ viewpoints /{guid}/comments/{guid} | DELETE |

|  |  |
| --- | --- |
| ***(Topics) (Viewpoints) Components*** | |
| /V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}/components  /V0.99/viewpoints /{guid}/components | GET,POST |
| V0.99/ teams/{guid}/projects/{guid}/topics/{guid}/ viewpoints /{guid}/components/{guid}  V0.99/components/{guid} | GET, DELETE |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Account**  **Owner** | **Project**  **Admin** | **Project**  **Editor** | **Project**  **Viewer** | **Model**  **Admin** | **Model**  **Editor** | **Model**  **Viewer** |
| **Create Project** | x | x |  |  |  |  |  |
| **Admin Project**  **(eg. invite)** | x | x |  |  |  |  |  |
| **Edit Project** | x | x | x |  |  |  |  |
| **Delete Project** | x | x |  |  |  |  |  |
| **View Project** | x | x | x | x | x | x | x |
|  |  |  |  |  |  |  |  |
| **Create Discipline** | x | x | x |  |  |  |  |
| **Edit Discipline** | x | x | x |  |  |  |  |
| **Delete Discipline** | x | x | x |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **View all Models** | x | x | x | x |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

|  |  |
| --- | --- |
| ***(Teams) Roles*** | |
| /V0.99/teams/{guid}/roles  [      {          "id": "f11d32e2-30b7-4f81-8a74-2165ecc00cf6",          "name": "Project\_Editor",          "customRole": false,          "resources": [              {                  "resource": "UserRightProject",                  "rights": [                      "Project\_Edit",                      "Project\_View",                      "Model\_ViewAll"                  ]}]},      {          "id": "2baca0e4-2eee-4f7c-bc56-22ed54a1859c",          "name": "Account\_Owner",          "customRole": false,          "resources": [              {                  "resource": "UserRightGlobal",                  "rights": [                      "AllProjects",                      "AllModels",                      "Project\_Create"                  ]}]      },      {          "id": "a298b28d-9711-4a76-9a7d-910cbf144ee5",          "name": "Project\_Admin",          "customRole": false,          "resources": [              {                  "resource": "UserRightProject",                  "rights": [                      "Project\_Admin",                      "Project\_Edit",                      "Project\_Delete",                      "Project\_View",                      "Model\_ViewAll",                      "Model\_Create"                  ]              }          ]      },      {          "id": "a618d075-7e4a-4bde-9d58-d2979696fa96",          "name": "Project\_Viewer",          "customRole": false,          "resources": [              {                  "resource": "UserRightProject",                  "rights": [                      "Project\_View",                      "Model\_ViewAll"                  ]}]}] | GET  ***Why is this team-related ?*** |
| ***(Projects) Members*** | |
| /V0.99/teams/{guid}/projects/{guid}/members  /V0.99/projects/{guid}/members  POST: Assign a project role to an existing member of a team. Only the user with "Account Owner" or "Project Owner" role can use this API call.  {    "member": {      "user\_id": "bf5b2382-1d14-b8df-8454-947f83b45c25"    },    role: {      id: "Project\_Editor"}    GET: Get the member list of a project along with their respective project roles.  [{  "member": {  "user\_id": "7aa846f2-6014-f68d-8dd4-4c7941d7cbcc",  "email": "customer1@bimplus.net"  },  "role": {  "id": "Project\_Editor" }},  {  "member": {  "user\_id": "bf5b2382-1d14-b8df-8454-947f83b45c25",  "email": "customer2@bimplus.net"  },  "role": {  "id": "Project\_Viewer“  }}]  PUT: Update the project role of a team member. Only the user with "Account Owner" or "Project Owner" role can use this API call.  {"member": {  "user\_id": "bf5b2382-1d14-b8df-8454-947f83b45c25",  "email": "customer@bimplus.net"  },  "role": {  "id": " Project\_Viewer"…… | GET,POST, PUT |

|  |  |
| --- | --- |
| ***Users*** | |
| /V0.99/users  POST: Create a new user | POST |
| /V0.99/users/{user\_id}  GET: Get information about the user  PUT: Modify the user  DELETE: Delete the user; Remove from all teams and projects | GET,PUT,DELETE |

|  |  |
| --- | --- |
| ***(Users) self*** | |
| /V0.99/users/self  GET all informations about the logged on user | GET |