|  |  |
| --- | --- |
|  | index.jpg |
|  |  |

BCF REST API

Version 0.99



© BuildingSMART

info@buildingsmart.com

<https://github.com/BuildingSMART/BCF-API>

[1. Introduction 3](#_Toc388002368)

[1.1 BCF REST API 3](#_Toc388002369)

[1.2 Authentication 3](#_Toc388002370)

[1.3 Paging, Sorting, Filtering 4](#_Toc388002371)

[1.4 Caching 5](#_Toc388002372)

[1.5 Cross origin resource sharing (Cors) 6](#_Toc388002373)

[2. Resources 7](#_Toc388002374)

[2.1 Information Resources 8](#_Toc388002375)

[2.1.1 schemas (link.json) 8](#_Toc388002376)

[2.1.2 colours (colour.json) 8](#_Toc388002377)

[2.2 Project Resources 9](#_Toc388002378)

[2.2.1 teams (team.json) 9](#_Toc388002379)

[2.2.2 projects (project.json) 10](#_Toc388002380)

[2.2.3 domains (domain.json) 11](#_Toc388002381)

[2.2.4 disciplines (discipline.json) 12](#_Toc388002382)

[2.3 Revision Resources 13](#_Toc388002383)

[2.3.1 revisions (revision.json) 13](#_Toc388002384)

[2.4 BCF Resources 14](#_Toc388002385)

[2.4.1 topics (topic.json) 14](#_Toc388002386)

[2.4.2 comments (comment.json) 15](#_Toc388002387)

[2.4.3 viewpoints (viewpoint.json) 16](#_Toc388002388)

[2.4.4 components (component.json) 17](#_Toc388002389)

[2.4.5 related\_topics (related\_topic.json) 17](#_Toc388002390)

[2.4.6 document\_references (document\_reference.json) 18](#_Toc388002391)

[2.5 User Resources 19](#_Toc388002392)

[2.5.1 Register 19](#_Toc388002393)

[2.5.1 Roles 19](#_Toc388002394)

1. Introduction

BCFv2 is the most recent version of BCF.

Github:

<https://github.com/BuildingSMART/BCF>

## 1.1 BCF REST API

BCF REST API v0.99 is the most recent version of the BCF REST API.

Github:

<https://github.com/BuildingSMART/BCF-API>

Examples:

<http://docs.bcfapi.apiary.io>

* 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

## 1.2 Authentication

OAuth2

## 1.3 Paging, Sorting, Filtering

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

**page**

**page\_size**

**sort**

Only „descending“ if property exits. If no descending property is there -> ascending=true

if descending=true -> descending

if descending=false -> ascending

**"&" to combine different query parameters (operations)**

Example:

*v2/projects/{guid}/topics?page=1&page\_size=5&sort=priority&descending=true*

**Filter use escape characters**  (<http://www.december.com/html/spec/esccodes.html>)

|  |  |  |
| --- | --- | --- |
| **Operator** | **Operation** | **Escape Characters** |
| > | Greater than | %3E |
| < | Less than | %3C |
| \* | Wild card for contains | \* |
| = | Equal to | %3D |
| >= | Greater than or equal to | %3E%3D |
| <= | Less than or equal to | %3C%3D |
| != | Is not equal to | %21%3D |
| & | And | %26 |
| | | Or | %7C |

Example:

*v2/projects/{guid}/topics?page=1&page\_size=5&sort=priority&descending=true&filter=(TopicLabel%3D Architecture%7CTopicLabel%3D Structural)%26TopicStatus%21%3DClosed*

## 1.4 Caching

ETags, or entity-tags, are an important part of HTTP, being a critical part of caching, and also used in "conditional" requests.

The ETag response-header field value, an entity tag, provides for an "opaque" cache validator.

The easiest way to think of an etag is as an MD5 or SHA1 hash of all the bytes in a representation. If just one byte in the representation changes, the etag will change.

ETags are returned in a response to a GET:

joe@joe-laptop:~$ **curl --include http://bitworking.org/news/**

HTTP/1.1 200 Ok

Date: Wed, 21 Mar 2007 15:06:15 GMT

Server: Apache

**etag: "078de59b16c27119c670e63fa53e5b51"**

Content-Length: 23081

…..

## 1.5 Cross origin resource sharing (Cors)

The server will put the "Access-Control-Allow-Headers" in the response header and specify who can access the server(json) resources. The client can look for this value and proceed with accessing the resources.

The server has a web config file .. "\*" means the server allow the resources for all domains.

<httpProtocol>

<customHeaders>

<add name="Access-Control-Allow-Headers" value="Content-Type, Accept, X-Requested-With, Authorization" />

<add name="Access-Control-Allow-Methods" value="GET, POST, PUT, DELETE, OPTIONS" />

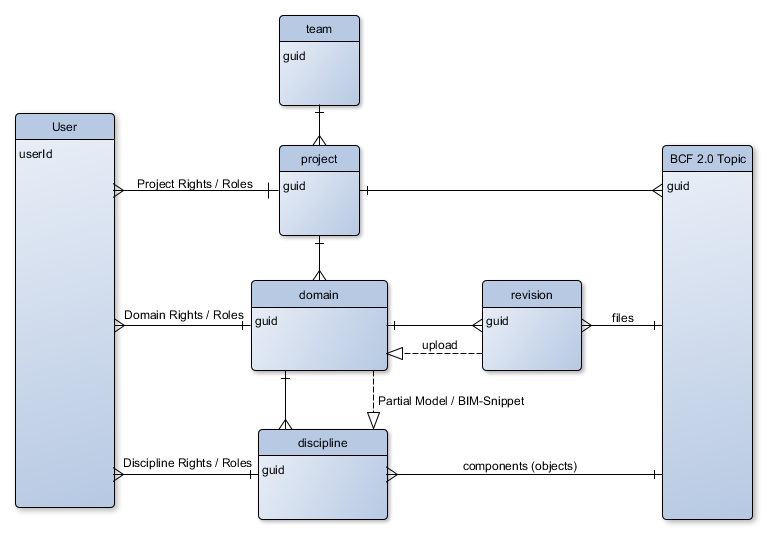
<add name="Access-Control-Allow-Origin" value="\*" />

</customHeaders>

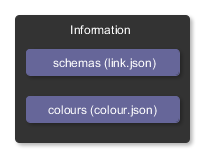
</httpProtocol>

2. Resources

* Information
* Project
* BCF
* Revision
* User



## 2.1 Information Resources



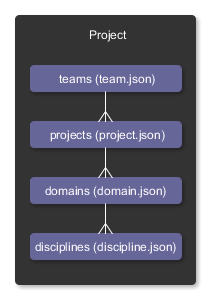
##### 2.1.1 schemas (link.json)

|  |
| --- |
| **GET** /V0.99/schemas   * GET - Retrive link schema that contains all available schemas in BCF-API. |
| **GET** /V0.99/schemas/{rel}   * GET - Retrive a specific schema |

##### 2.1.2 colours (colour.json)

|  |
| --- |
| **GET** /V0.99/colours   * GET - Retrive ARGB values for colours. |

## 2.2 Project Resources



##### 2.2.1 teams (team.json)

|  |
| --- |
| **GET, POST** /V0.99/teams   * GET - Retrieve a list of teams where the currently logged on user is assigned to with his specific roles. * POST - Add a new team |
| **GET, PUT, DELETE** /V0.99/teams/{guid}   * GET - Retrieve a specific team * PUT - Modify a specific team * DELETE - Delete a specific team |

##### 2.2.2 projects (project.json)

|  |
| --- |
| **GET, POST** /V0.99/ teams/{guid}/projects   * GET - Retrieve a list of projects where the currently logged on user is assigned to with his specific roles. * POST - Add a new project |
| **GET, PUT, DELETE** /V0.99/projects/{guid}   * GET - Retrieve a specific project * PUT - Modify a specific project * DELETE - Delete a specific project   Long URL: /V0.99/ teams/{guid}/projects/{guid} |
| **GET, PUT, DELETE** /V0.99/projects/{guid}/thumbnail   * GET - Retrieve the project thumbnail * PUT - Add, change the project thumbnail * DELETE – Delete the project thumbnail   Long URL: /V0.99/ teams/{guid}/projects/{guid}/thumbnail |
| **GET, PUT, DELETE** /V0.99/projects/{guid}/extensions   * GET - Retrieve the project extension schema (extensions.json) * PUT - Add, change the project extension schema (extensions.json) * DELETE – Delete the project extension schema (extensions.json)   Long URL: /V0.99/ teams/{guid}/projects/{guid}/extensions |

##### 2.2.3 domains (domain.json)

|  |
| --- |
| **GET, POST** /V0.99/projects/{guid}/domains   * GET - Retrieve a list of domains within a project where the currently logged on user is assigned to with his specific roles. * POST - Add a new domain to a project   Long URL: /V0.99/ teams/{ guid }/projects/{guid}/domains |
| **GET, PUT, DELETE** /V0.99/domains/{guid}   * GET - Retrieve a specific domain * PUT - Modify a specific domain * DELETE - Delete a specific domain   Long URL: /V0.99/ teams/{ guid }/projects/{guid}/domains/{guid} |

##### 2.2.4 disciplines (discipline.json)

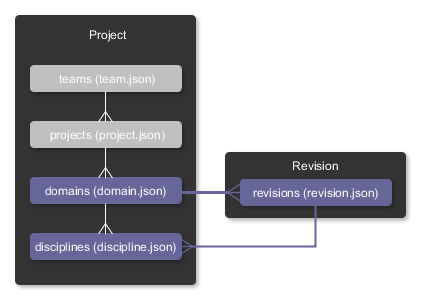
All *Disciplines* in BCF-API are predefined. By uploading a *Revision* (IFC-File) the server detects all *Disciplines* that the *Revision* contains. If it detects a *Discipline* which is not within the *Domain* the *Revision* was uploaded, a BCF-Topic automatically will be created.

Further description needed ........ also to add is the list of Disciplines here.........

Also a description of objects related to disciplines is needed

|  |
| --- |
| **GET, POST** /V0.99/domains/{guid}/disciplines   * GET - Retrieve a list of disciplines within a domain. * POST - Add a new discipline (from predefined list) to a domain   Long URL: /V0.99/ teams/{ guid }/projects/{guid}/domains/{guid}/diciplines |
| **GET, PUT, DELETE** /V0.99/discipline/{guid}   * GET - Retrieve a specific discipline * PUT - Modify a specific discipline * DELETE - Delete a specific discipline   Long URL: /V0.99/ teams/{ guid }/projects/{guid}/domains/{guid}/disciplines/{guid} |

## 2.3 Revision Resources

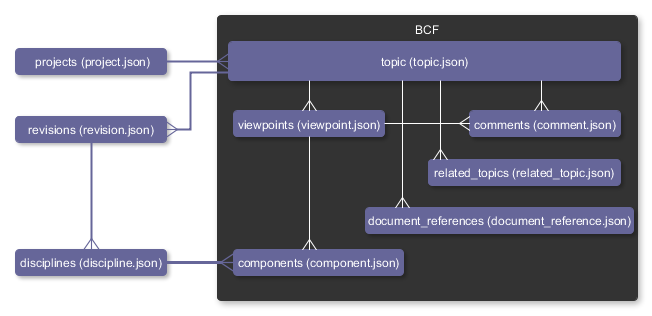


Revisions (IFC-Files) are belonging to a *Domain* where they are uploaded to. A *Revision* can contain multiple *Disciplines* (see 2.2.4).

##### 2.3.1 revisions (revision.json)

|  |
| --- |
| **GET, POST** /V0.99/domains/{guid}/revisions   * GET - Retrieve revisions of a domain. * POST - Add (Upload) a new revision to a domain   Long URL: /V0.99/ teams/{guid }/projects/{guid}/domains/{guid}/revisions |
| **GET, DELETE** /V0.99/revisions/{guid}   * GET - Retrieve a specific revision * DELETE - Delete a specific revision   Long URL: /V0.99/ teams/{guid }/projects/{guid}/domains/{guid}/revisions/{guid} |

## 2.4 BCF Resources



##### 2.4.1 topics (topic.json)

|  |
| --- |
| **GET, POST** /V0.99/projects/{guid}/topics   * GET - Retrieve topics of a project (default sort = CreationDate) * POST - Add a new topic to a project   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics |
| **GET, PUT, DELETE** /V0.99/topics/{guid}   * GET - Retrieve a specific topic * PUT - Modify a specific topic * DELETE - Delete a specific topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid} |
| **GET** /V0.99/topics/{guid}/revisions   * GET - Retrieve all revisions related to a topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/revisions |

##### 2.4.2 comments (comment.json)

|  |
| --- |
| **GET, POST** /V0.99/topics/{guid}/comments   * GET - Retrieve comments of a topic * POST - Add a new comment to a topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/comments |
| **GET, PUT, DELETE** /V0.99/comments/{guid}   * GET - Retrieve a specific comment * PUT - Modify a specific comment * DELETE - Delete a specific comment   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/comments/{guid} |
| **GET, POST, DELETE** /V0.99/comments/{guid}/viewpoint   * GET - Retrieve the viewpoint related to a comment * POST - Add a viewpoint relation to a comment * DELETE - Delete the viewpoint relation on a comment   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/comments/{guid}/viewpoints |
| **GET, POST, DELETE** /V0.99/comments/{guid}/reply\_to   * GET - Retrieve the replyTo comment related to a comment * POST - Add a replyTo comment relation to a comment * DELETE - Delete the replyTo comment relation on a comment   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/comments/{guid}/reply\_to |

##### 2.4.3 viewpoints (viewpoint.json)

|  |
| --- |
| **GET, POST** /V0.99/topics/{guid}/viewpoints   * GET - Retrieve viewpoints of a topic * POST - Add a new viewpoint to a topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/viewpoints |
| **GET, PUT, DELETE** /V0.99/viewpoints/{guid}   * GET - Retrieve a specific viewpoint * PUT - Modify a specific viewpoint * DELETE - Delete a specific viewpoint   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/viewpoints/{guid} |
| **GET, POST** /V0.99/viewpoints/{guid}/comments   * GET - Retrieve all comments related to a viewpoint   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/viewpoints/{guid}/comments |
| **GET, POST, DELETE** /V0.99/viewpoints/{guid}/bitmap   * GET - Retrieve the bitmap related to a viewpoint * POST - Add a bitmap to the viewpoint * DELETE - Delete the bitmap of the viewpoint   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/viewpoints/{guid}/bitmap |

##### 2.4.4 components (component.json)

|  |
| --- |
| **GET, POST** /V0.99/viewpoints/{guid}/components   * GET - Retrieve components of a viewpoint * POST - Add a new component to a viewpoint   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/viewpoints/{guid}/components |
| **GET, PUT, DELETE** /V0.99/components/{guid}   * GET - Retrieve a specific component * PUT - Modify a specific component * DELETE - Delete a specific component   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/viewpoints/{guid}/components/{guid} |

##### 2.4.5 related\_topics (related\_topic.json)

|  |
| --- |
| **GET, POST** /V0.99/topics/{guid}/related\_topics   * GET - Retrieve related topics of a topic * POST - Add a new related\_topic to a topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/related\_topics |
| **DELETE** /V0.99/topics/{guid}/related\_topics/{guid}   * DELETE - Delete a related topic from a topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/related\_topics/{guid} |

##### 2.4.6 document\_references (document\_reference.json)

|  |
| --- |
| **GET, POST** /V0.99/topics/{guid}/document\_references   * GET - Retrieve documents referenced of a topic * POST - Add a new document reference to a topic   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/document\_references |
| **GET, DELETE** /V0.99/topics/{guid}/document\_references/{guid}   * GET - Retrieve a document referenced to a topic * DELETE - Delete a document reference   Long URL: /V0.99/teams/{guid}/projects/{guid}/topics/{guid}/document\_references/{guid} |

## 2.5 User Resources

....

##### 2.5.1 Register

....

##### 2.5.1 Roles

.....