

Blueprint REST API

Developer Guide

Blueprint REST API

Overview

The Blueprint REST API allows you to query data in your Blueprint projects. The API is based on REST principles, making it easy to write and test applications. You can use your browser to access URIs, or you can use an HTTP client in any programming language to interact with the API.

System Requirements

Before you begin, ensure the following system requirements are met:

- You must have Blueprint version 5.2 or later.

Note: Some API services require Blueprint 5.4 or later.

- You must have an API license. To enable API access, please login to the Customer Portal (<http://portal.blueprintsys.com>) and submit a support case.
- You must have a Blueprint username and password.

About REST (REpresentational State Transfer)

REST-style architectures consist of clients and servers:

1. Clients initiate requests to servers
2. Servers process requests and return appropriate responses.

Requests and responses are built around the transfer of representations of resources. The client begins sending requests when it is ready to make the transition to a new state. While one or more requests are outstanding, the client is considered in transition. The representation of each application state contains links that may be used the next time the client chooses to initiate a new state-transition.

Requests

Objective	Description
Authenticate	Generates an authentication token for the specified user.
List Projects	Retrieves a list of projects.
Get Project	Retrieves a project by ID, and returns information about the project.
Create Project	Creates an empty project from a template or without a template.
List Artifact Types	Retrieves all artifact types available in the specified project.

Objective	Description
Get Artifact Type	Retrieves all artifact type objects from a project.
List Artifacts	Retrieves a list of artifacts. It is also possible to filter requested artifacts.
Get Artifact	Retrieves a single artifact by Project ID and Artifact ID and returns information about the artifact.
Get Child Artifacts Of Artifact	Retrieves the child artifact(s) of the specified artifact.
Get Root Artifacts Of Project	Retrieves the artifact(s) at the root level of the project.
Add Artifact	Adds a single artifact to the specified project. This service is supported in Blueprint version 5.4 and later.
Update Artifacts	Updates properties of the specified artifact(s).
Publish Artifacts	Publishes changes of the specified artifact(s).
Discard Artifacts	Discards changes to the specified artifact(s).
Get Group	Retrieves the group by its ID.
List Groups	Retrieves a list of all groups in this Blueprint instance.
List Users	Retrieves all users in this Blueprint instance.
Get User	Retrieves the user with the specified user ID.
Create User	Creates a user with specific defined properties.
Update User	Updates the specified properties of a specific user.
Delete User	Deletes all specified users.
Delete Artifact	Deletes the specified artifact. You can also delete all children of the specified artifact. This service is supported in Blueprint version 5.4 and later.
Add Attachment	Adds a single file attachment to the specified artifact. This service is supported in Blueprint version 5.4 and later.
Add Attachment to Subartifact	Adds a single file attachment to the specified subartifact. This service is supported in Blueprint version 5.4 and later.
Delete Attachment	Deletes a single file attachment from an artifact. This service is supported in Blueprint version 5.4 and later.
Get Attachments	Retrieves attachment content. This service is supported in Blueprint version 5.4 and later.
Add Comment	Adds a single comment to the specified artifact. This service is supported in Blueprint version 5.4 and later.
Delete Comment	Deletes the specified comment from an artifact. This service is supported in Blueprint version 5.4 and later.

Objective	Description
Update Comment	Updates the specified comment. This service is supported in Blueprint version 5.4 and later.
Publish Comments	Publishes all unpublished comments associated with the specified artifact. This service is supported in Blueprint version 5.4 and later.
Rate Comment	Rates a comment by specifying whether you like or dislike the comment. This service is supported in Blueprint version 5.4 and later.
Reply to Comment	Reply to the specified comment. This service is supported in Blueprint version 5.4 and later.
Rate Reply	Rates a reply by specifying whether you like or dislike the reply. This service is supported in Blueprint version 5.4 and later.
Delete Reply	Deletes the specified reply from the specified artifact comment. This service is supported in Blueprint version 5.4 and later.
Update Reply	Updates the specified reply. This service is supported in Blueprint version 5.4 and later.
Get Artifact Image	Retrieves the main image from the specified graphical artifact in PNG format. This service is supported in Blueprint version 5.4 and later.
Get Blueprint Product Version	Retrieves the Blueprint product version. This service is supported in Blueprint version 5.4 and later.
Get Project Discussion Status Values	Retrieves a list of discussion status values that are valid for the specified project. This service is supported in Blueprint version 5.4 and later.
Add Traces	Adds trace(s) to the artifact you specify. This service is supported in Blueprint version 5.4 and later.
Move Artifact	In order to <i>move</i> an artifact, you must change the parent of the artifact. By changing the parent, you are effectively moving the specified artifact and its descendants to a new location in the hierarchy. This service is supported in Blueprint version 5.4 and later.
Delete Traces	Deletes the specified trace(s). This service is supported in Blueprint version 5.4 and later.
List Unpublished Artifacts	Retrieves a list of artifacts with unpublished changes. This service only returns artifacts in the list if the changes were made by the current user. This service is supported in Blueprint version 5.4 and later.
List Reviews	Retrieves a list of reviews by Project ID. This service is supported in Blueprint version 5.5.1 and later.
Get Review	Retrieves a review by Project ID and Artifact ID and returns information about the review. This service is supported in Blueprint version 5.5.1 and later.
List ALM Targets	Retrieves a list of ALM targets. This service is supported in Blueprint version 5.5.1 and later.
Get ALM Job	Retrieves an ALM job by Project ID, Target ID and Job ID and returns information about the job. This service is supported in Blueprint version 5.5.1 and later.

Objective	Description
Add ALM Job	Adds an ALM job to the specified project using the specified target. This service is supported in Blueprint version 5.5.1 and later.
Get Collection	Retrieves the collection in the specified project. This service is supported in Blueprint version 6.0 and later.
List Collections	Lists collections in the specified project. This service is supported in Blueprint version 6.0 and later.

Security and authentication

Blueprint's REST API is served over HTTPS if Blueprint is configured to run with SSL. HTTPS ensures secure communication of API requests and responses.

Blueprint exclusively supports Basic Authentication for the submission of credentials. Thereafter, a proprietary token is used. To obtain a token, you must provide an Authorization header with Basic scheme. Refer to the [Quick Start Example](#) for more information.

Token expiry

You can obtain the token expiry by viewing the header of the Authenticate response. The token expiry is stored in the `blueprinttokenexpirydate` parameter of the Authenticate response header.

Here is an example of the `blueprinttokenexpiry` parameter in the [Authenticate](#) response header:

```
{
  'content-length': '238',
  'x-xss-protection': '1; mode=block',
  'blueprinttokenexpirydate': 'Thu, 28 Nov 2013 17:20:34 GMT',
  'x-content-type-options': 'nosniff',
  'expires': '-1',
  'server': 'Blueprint',
  'pragma': 'no-cache',
  'cache-control': 'no-cache',
  'date': 'Thu, 28 Nov 2013 16:50:34 GMT',
  'x-frame-options': 'deny',
  'content-type': 'application/json; charset=utf-8',
  'authorization': 'BlueprintToken M6Uf+rS0WsqD89d2Rf+nw18...
[Truncated]...GtqNUWYFuNC81mF0o='
}
```

Requests and responses

Submission methods

API requests can be submitted using the following HTTP methods:

- HEAD
- GET
- POST
- PATCH
- DELETE

Supported formats

Response formats

The Blueprint REST API supports the following response formats:

- JSON (default)

To use JSON format, add the following information to your request header:

```
Accept: application/json
```

- XML

To use XML format, add the following information to your request header:

```
Accept: application/xml
```

Request formats

The Blueprint REST API supports the following request formats:

- JSON

To use JSON format for your request body, add the following information to your request header:

```
Content-Type: text/json
```

- XML

To use XML format for your request body, add the following information to your request header:

```
Content-Type: text/xml
```

Note: XML data in the request body must adhere to the [Blueprint REST API XML Schema](https://documentation.blueprintcloud.com/DeveloperCenter/RestAPI/Content/Schema/Blueprint_XMLSchema_RESTAPI_55.xsd) (https://documentation.blueprintcloud.com/DeveloperCenter/RestAPI/Content/Schema/Blueprint_XMLSchema_RESTAPI_55.xsd).

Pagination

Blueprint limits the number of results that can be returned in a single response in order to ensure optimal performance. Paging is controlled using the `Offset` and `Limit` parameters:

- `Offset` (optional): Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- `Limit` (optional): Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

When pagination is in effect, response will always be returned with an HTTP status code of **206: Partial Content**. When this happens, the response header always contains a `content-range` parameter that indicates the items included in the response and the total number of matches. If the total number of matches is unknown, an asterisk appears instead. For example, the following response header indicates that there are a total of 122 users and the response includes users at position 13 to 15.

```
{'content-length': '822', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'users 13-15/122', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 11:59:23 GMT', 'content-type': 'application/json; charset=utf-8'}
```

Pagination Examples

This request returns the first 50 artifacts (that is, artifacts 0 - 49):

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts?Offset=0&Limit=50
```

This request returns the next 100 artifacts (that is, artifacts 50 - 149):

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts?Offset=50&Limit=100
```

This request returns the next 25 artifacts (that is, artifacts 150 - 174):

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts?Offset=150&Limit=25
```

HTTP status codes

- **200**: OK
- **201**: Successfully Created
- **206**: Partial Content
- **400**: Bad Request

This error can occur for a variety of reasons:

- At least one URI or request body parameter is invalid, such as one of the [Filter](#), [Traces](#), [Offset](#) or [Limit](#) parameters.
- A required value has not been provided.
- If you are submitting a [POST](#) or [PATCH](#) request and have not added the `X-HTTP-Method-Override` parameter to your request header.
- **401: Unauthorized**
 - The Blueprint token is missing, invalid or expired.
- **403: Forbidden**
 - The applicable permissions are needed. For example, this error can occur when:
 - An Author or Collaborate license is needed for access.
 - Comment permissions are needed on the artifact.
 - Edit permissions are needed on the change summary job.
- **404: NotFound**
 - Item is not found in the project. For example, this error can occur when:
 - Artifact not found.
 - Artifact type not found.
 - Project not found.
 - ALM target not found.
 - Change summary job not found.
 - User does not have edit permissions for this artifact, project or ALM target.
- **409: Conflict**
 - Reason for failure can be found in the response body. Additional information may also be found in the log.
- **411: Length Required**
 - If the Content-Length header is missing in the request.
- **413: Request Entity Too Large**
 - If the provided file size is larger than allowed.
- **415: Unsupported Media Type**
 - If the provided media type is not of MIME Multipart Content "form-data."
- **416: Request Range Not Satisfiable**
 - If `offset` points to a part of a collection that is not available.
- **500: Internal Server Error**
- **501: Not Implemented**

Resources

Below are some helpful resources:

- [Blueprint XML Schema](http://documentation.blueprintcloud.com/DeveloperCenter/RestAPI/Content/Schema/Blueprint_)
(http://documentation.blueprintcloud.com/DeveloperCenter/RestAPI/Content/Schema/Blueprint_)

XMLSchema_RESTAPI_55.xsd)

- <http://www.w3schools.com/schema/>
Provides information about XML schemas.
- <http://tools.ietf.org/html/rfc2617>
Provides information about HTTP authentication.
- <http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>
Provides information about HTTP status codes.
- <http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html>
Provides information about HTTP response header parameters.

Quick Start Example

In this tutorial, we're going to walk through some sample code that shows you how to authenticate using the API. This example will demonstrate how you can request a token and then submit a request to obtain a list of Blueprint projects.

Request a token

We'll start by obtaining a token by submitting an [Authenticate](#) request.

The authentication request must be sent to `http://today/authentication/v1/login` using the GET method. The request header must contain an `Authorization` header in the following format:

```
Authorization: Basic <EncodedCredentials>
```

... where `<EncodedCredentials>` is the Base 64 encoded string for the credentials in the following format:

```
<username>:<password>
```

Example

If the username is `jsmith` and the password is `Pa55w0rd`, you must calculate the Base 64 encoded string for:

```
jsmith:Pa55w0rd
```

The Base 64 encoded string is `anNtaXRoOIBhNTV3MHJk`, and therefore the authorization header looks like this:

```
Authorization: Basic anNtaXRoOIBhNTV3MHJk
```

If your credentials are correct, the response contains the access token that you can use to submit API requests. If the `Authorization` header is not formatted correctly or if your credentials are incorrect, the service returns a **401 Unauthorized call** error.

You can obtain the token expiry by viewing the header of the Authenticate response. The token expiry is stored in the `blueprinttokenexpirydate` parameter of the Authenticate response header.

Python Example

The following code demonstrates how you can obtain an access token using Python:

```
def get_token():
    # Of course, we wouldn't normally hardcode credentials
    username = "api_docs_user"
    password = "Pa55w0rd"

    # concatenate the username and password with : as required by rfc2617
    auth_string = username + ':' + password
    # encode in utf-8 and perform base64 encoding
    auth_byte = auth_string.encode('utf-8')
    auth_encode = base64.b64encode(auth_byte)
    # decode in utf-8 string for the header
```

```
auth = auth_encode.decode('utf-8')

request_uri = 'https://production.blueprintcloud.com/authentication/v1/login'
request_header={
    'Authorization' : 'Basic ' + auth,
    'Accept' : 'application/json'
}

# make the request using GET
response = requests.get(request_uri, headers=request_header)
token = response.json()
return token
```

Submit request

With a request token in hand, you can now submit a request (example: [List Projects](#)) to obtain data from Blueprint. You must include your access token in the `Authorization` header of every request.

The authorization header must look like this:

```
Authorization: BlueprintToken <TokenValue>
```

...where `<TokenValue>` is the token obtained from the authentication call.

Example

```
Authorization: BlueprintToken 3L7b+nMqXlpE/dBptnomyZiZpZc
[...Truncated...]a6cpdsCS+uWMIB5fODHItoo0fCiwP
```

Python Example

The following code demonstrates how you can obtain a list of Blueprint projects using Python:

```
def list_projects():
    # obtain a token from our get_token() function
    token = get_token()

    # build the resource URI
    resource_uri = 'https://production.blueprintcloud.com/api/v1/projects'

    # build the request header
    request_header={
        'Authorization' : 'BlueprintToken ' + token,
        'Accept' : 'application/json'
    }

    # specify the URI parameters
    uri_parameters_list={
        'Location' : 'True'
    }

    # create the request uri
    request_uri = resource_uri + '?' + urllib.parse.urlencode(uri_parameters_list)
    output_requesturi(request_uri)
```

```
# submit the request using HTTP GET method
response = requests.get(request_uri, headers=request_header)
return response
```

Parse the result

After you receive a response, you will need to parse the XML or JSON data. For example, if you requested a list of projects, you will probably want to store the project data in a python list.

Authenticate

For Blueprint version 5.2 and 5.3, use the following:

```
/authentication/v1/login
```

For Blueprint version 5.4 and later, use the following:

```
/authentication/v1/loginEx
```

DESCRIPTION Generates an authentication token for the specified user.

Blueprint exclusively supports Basic Authentication for the submission of credentials. Thereafter, a proprietary token is used. To obtain a token, you must provide an Authorization header with Basic scheme. Refer to the [Quick Start Example](#) for more information.

Important: To enable API access, please login to the Customer Portal (<http://portal.blueprintsys.com>) and submit a support case.

URL STRUCTURE

For Blueprint version 5.2 and 5.3, use the following:

```
[Blueprint_URI]/authentication/v1/login
```

For Blueprint version 5.4 and later, use the following:

```
[Blueprint_URI]/authentication/v1/loginEx
```

SUPPORTED METHODS ■ [GET](#)

Notes

Tip

You can obtain the token expiry by viewing the header of the Authenticate response. The token expiry is stored in the `blueprinttokenexpirydate` parameter of the Authenticate response header.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/authentication/v1/login
```

Response Header

```
{'content-length': '313', 'x-xss-protection': '1; mode=block',  
'blueprinttokenexpirydate': 'Thu, 28 Nov 2013 17:47:20 GMT', 'x-content-type-  
options': 'nosniff', 'expires': '-1', 'server': 'Blueprint', 'pragma': 'no-  
cache', 'cache-control': 'no-cache', 'date': 'Thu, 28 Nov 2013 17:17:20 GMT',  
'x-frame-options': 'deny', 'content-type': 'application/xml; charset=utf-8',  
'authorization': 'BlueprintToken  
c5vEipeBiH0mngOpIrnTj+dVSHgWzLGm3XE/OBk8JJ67D2lQnhQneAeKi0GLoLw+lqDQAaCQg0V0Hc7  
qcmp+FHbkaHSK6/yrzF10MNz64VHXc4uTJjpe91XFhL1B+yBBexRVtydzVgw4zrjRxT5qYwosGr/jfi  
9hR82khFYIolnzzUeQ+wUj95337u3hDtyINqGdY2Qj/IlEzaToo9dQajX4fZoH3HKOPG/BtH1JXwc  
='}
```

Response Body

```
<?xml version="1.0" encoding="UTF-8"?>  
<AuthenticationInfo xmlns="http://www.blueprintsys.com/blueprint/api/v1"  
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">  
  
<Token>XEjna6FKp734EH0fyJ3T0rIeVuBqejQWZAULgRzsz4IpfPKXrZYbk6sGijD+xRjK1grgAe71  
bBA6768BotXCWVTfgDdhyZiV7dsGIeu4UXJ55SuWDpnsW4uViCbSdENdqD5b7JOKES98KfdDfFK+chG  
MUBw3uN6I20nKoehJ8KWTMz+4o077yMrtknZjqoTbto5wrfnNND+IndZk0Lzqu7RMvHdHtjPxbkHeZ1  
TcaOA=</Token>  
  <TokenExpirationTime>2013-12-05T14:55:14.9131132Z</TokenExpirationTime>  
  <UserDisplayName>Default Instance Admin</UserDisplayName>  
  <UserId>1</UserId>  
  <UserLicense>Author</UserLicense>  
  <UserName>admin</UserName>  
</AuthenticationInfo>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/authentication/v1/login
```

Response Header

```
{'content-length': '238', 'x-xss-protection': '1; mode=block',  
'blueprinttokenexpirydate': 'Thu, 28 Nov 2013 17:20:34 GMT', 'x-content-type-  
options': 'nosniff', 'expires': '-1', 'server': 'Blueprint', 'pragma': 'no-  
cache', 'cache-control': 'no-cache', 'date': 'Thu, 28 Nov 2013 16:50:34 GMT',  
'x-frame-options': 'deny', 'content-type': 'application/json; charset=utf-8',  
'authorization': 'BlueprintToken  
M6Uf+rS0WsqD89d2Rf+nw18g/RPoC4T01Iecy0FxFK+vJaoYJErY3Dc9zGuk08wQq93yYkx+v/j7S1Ko  
oxYY/SE7/e7XGAYV5A5AABtw0hH/lbvtZcvjdaDB0t91faamSQFuRz5PVQHHvktmais92QdFn/Eu8C  
WRYbLazMHEcVTT9HNMENVDLvUJl33+WIKw9XticLuiIogQh9rgRxd0WyBPvEGtqNUWYFuNC81mF0o  
='}
```

Response Body

```
{  
  "Token":  
  "dQ5eXjxQ2TbpJTgTo0EknQXwJceOPfBVVGQNiPB6aRmdIc08AWwxEnfpXZ1N/k04W8aB+uScCy3bs3  
  IDmYIcfBdnuH7YysSl41BuiM1+4oCNPkGKhb6YuZy83018sOtMjIExLzdPG5422BW/JwRtpwPbRhFzX  
  iu9aoTgH4wB9HkS/XWhl3Ki5IDgABT9OFMwCsviwhSEjMOTHPEkJeEls5ophGkK4HTR2htZDSZdx8E  
  =",  
  "TokenExpirationTime": "2013-12-05T14:54:49.1166224Z",  
  "UserName": "admin",  
  "UserDisplayName": "Default Instance Admin",  
  "UserId": 1,  
  "UserLicense": "Author"  
}
```

List Projects

```
/api/v1/projects?location={location}&offset={offset}&limit={limit}
```

DESCRIPTION Retrieves a list of projects.

Projects are only included in the response if the user (that is was authenticated to obtain the token) has permissions to view those projects.

URL STRUCTURE [Blueprint_URI]/api/v1/projects

SUPPORTED METHODS

- HEAD
- GET

REQUEST URI PARAMETERS

- **Location (optional)**: Indicates whether to retrieve comments of the artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **Offset (optional)**: Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional)**: Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects?Location=True
```

Response Header


```
{'content-length': '761', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 14:58:51 GMT', 'expires': '-1', 'content-type':  
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>  
<Projects xmlns="http://www.blueprintsys.com/blueprint/api/v1"  
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">  
  <Project>  
    <Description>Documentation planning and collaboration.</Description>  
    <Id>13691</Id>  
    <Location>/Blueprint/Product Management/Steph/Documentation</Location>  
    <Name>Documentation</Name>  
  </Project>  
  <Project>  
    <Description>This project is for the creation of an online portal for BP  
Airways customers to make reservations and other related  
activities.</Description>  
    <Id>220870</Id>  
    <Location>/Blueprint/Product Management/Steph/BP Airways</Location>  
    <Name>BP Airways</Name>  
  </Project>  
  <Project>  
    <Description/>  
    <Id>221280</Id>  
    <Location>/Blueprint/Product Management/Steph/Online Banking</Location>  
    <Name>Online Banking</Name>  
  </Project>  
</Projects>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects?Location=True
```

Response Header

```
{'content-length': '516', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 10:47:30 GMT', 'expires': '-1', 'content-type':  
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[  
  {  
    "Location": "/Blueprint/Product Management/Steph/Documentation",  
    "Description": "Documentation planning and collaboration.",  
    "Id": 13691,  
  }  
,  
  {  
    "Location": "/Blueprint/Product Management/Steph/BP Airways",  
    "Description": "This project is for the creation of an online portal for BP  
Airways customers to make reservations and other related  
activities.",  
    "Id": 220870,  
  }  
,  
  {  
    "Location": "/Blueprint/Product Management/Steph/Online Banking",  
    "Description": "",  
    "Id": 221280,  
  }  
]
```

```
    "Name": "Documentation"
  },
  {
    "Location": "/Blueprint/Product Management/Steph/BP Airways",
    "Description": "This project is for the creation of an online portal for BP Airways customers to make reservations and other related activities.",
    "Id": 220870,
    "Name": "BP Airways"
  },
  {
    "Location": "/Blueprint/Product Management/Steph/Online Banking",
    "Description": "",
    "Id": 221280,
    "Name": "Online Banking"
  }
]
```

Get Project

```
/api/v1/projects/{projectId}
```

DESCRIPTION Retrieves a project by ID, and returns information about the project.

URL [Blueprint_URI]/api/v1/projects/{ProjectId}

STRUCTURE

- ProjectId: The ID of the project.

SUPPORTED METHODS

- HEAD
- GET

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870?
```

Response Header

```
{'content-length': '384', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 14:58:52 GMT', 'expires': '-1', 'content-type':  
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>  
<Project xmlns="http://www.blueprintsys.com/blueprint/api/v1"  
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">  
  <Description>This project is for the creation of an online portal for BP  
  Airways customers to make reservations and other related  
  activities.</Description>
```

```
<Id>220870</Id>
<Location>/Blueprint/Product Management/Steph/BP Airways</Location>
<Name>BP Airways</Name>
</Project>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870?
```

Response Header

```
{'content-length': '238', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 10:47:32 GMT', 'expires': '-1', 'content-type':
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
{
  "Location": "/Blueprint/Product Management/Steph/BP Airways",
  "Description": "This project is for the creation of an online portal for BP Airways customers to make reservations and other related activities.",
  "Id": 220870,
  "Name": "BP Airways"
}
```

Create Project

```
/api/v1/projects
```

DESCRIPTION Creates an empty project from a template or without a template.

URL [Blueprint_URI]/api/v1/projects
STRUCTURE

SUPPORTED METHODS ■ POST

REQUEST BODY The new project must be defined in the body of the request.

PARAMETERS

- Name (required): Defines the name of the new project.
- Description (optional): Defines the description of the new project.
- Path (optional): Defines the location of the new project. This parameter must be the full path to an instance folder. If not provided, the newly-created project will be added to the root.
- TemplateLocation (optional): Defines the template used for the new project. This parameter must be the full path of the template project. If not provided, the newly-created project will not be based off of an existing project.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects
```

Request Body

```
<?xml version="1.0"?>
<CreateProjectRequest
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Description>This is my new Project.</Description>
  <Name>New Project</Name>
  <Path>/Blueprint</Path>

<TemplateLocation>/Blueprint/templates/DefaultProjectTemplate</TemplateLocation>
</CreateProjectRequest>
```

Response Body

```
<?xml version="1.0" ?>
<Project
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Description>This is my new project.</Description>
  <Id>4344</Id>
  <Location>/Blueprint/New Project</Location>
  <Name>New Project</Name>
</Project>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects
```

Request Body

```
[
  {
    "Name": "New Project",
    "Description": "This is my new project.",
    "Path": "/Blueprint",
    "TemplateLocation": "/Blueprint/templates/DefaultProjectTemplate"
  }
]
```

Response Body

```
[
  {
    "Id": 4344,
    "Name": "New Project",
  }
]
```

```
    "Description": "This is my new project.",  
    "Location": "/Blueprint/New Project"  
  }  
]
```

List Artifacts

```
/api/v1/projects/{projectId}/artifacts?properties={properties}&status={status}&comments={comments}&traces={traces}&attachments={attachments}&richtextasplain={richtextasplain}&inlinecss={inlinecss}&offset={offset}&limit={limit}&content={content}
```

DESCRIPTION Retrieves a list of artifacts. It is also possible to [filter](#) requested artifacts. Artifacts are only included in the response if the user (that is was authenticated to obtain the token) has view permissions to those artifacts.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts

- **ProjectId**: The ID of the project.

SUPPORTED METHODS

- **HEAD**
- **GET**
- **POST**: If you want to filter the results, you must use override the GET method and use the POST method instead.

REQUEST URI PARAMETERS

- **Properties (optional)**: Defines whether the properties should be loaded for the artifacts. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **Status (optional)**: Defines whether or not the status of the artifact should be loaded. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **Comments (optional)**: Indicates whether to retrieve comments of the artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **Traces (optional)**: Indicates whether to retrieve traces of the artifact. Valid options include: **None**, **All**, **Parent**, **Child**, **Manual**, **Reuse** and **Other**. The default is **None** if not specified. The default is **All** if the parameter is included in the URI with no given value.
- **Attachments (optional)**: Indicates whether to retrieve information about the attachments of the artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **RichTextAsPlain (optional)**: Defines whether or not to retrieve all rich-text

properties as Plain Text instead of HTML. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

- **InlineCSS (optional)**: Defines whether or not to retrieve all rich-text properties with locally defined or inline styles. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value. When this parameter is set to **false**, rich-text properties return internal styles that are defined within the `<head>` section of the HTML.

Note: This parameter is supported in Blueprint 6.1 and later.

- **Offset (optional)**: Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional)**: Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.
- **Content (optional)**: Defines whether or not to retrieve the artifact's content. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/artifacts?traces=all&attachments=true
```

Request Body

```
filter={artifact-type eq 'Business Process Diagram';}
```

Response Header

```
{'content-length': '2889', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 14:58:54 GMT', 'expires': '-1', 'content-type':  
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>  
<Artifacts  
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <Artifact>  
    <ArtifactTypeId>317881</ArtifactTypeId>  
    <ArtifactTypeName>Business Process Diagram</ArtifactTypeName>  
    <Attachments>  
      <Attachment>  
        <FileName>Contract_of_Carriage.pdf</FileName>  
        <Id>103544</Id>  
      </Attachment>  
    </Attachments>  
    <Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/44  
05717/attachments/103544</Link>  
  </Artifact>  
  <BaseArtifactType>BusinessProcess</BaseArtifactType>  
  
<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405717</Blueprin  
tUrl>  
  <Id>4405717</Id>  
  <Name>Traveler Booking</Name>  
  <ParentId>4405707</ParentId>  
  <ProjectId>4405707</ProjectId>  
  <Traces>  
    <Trace>  
      <ArtifactId>4405788</ArtifactId>  
      <ArtifactPropertyName>Traveler Booking</ArtifactPropertyName>  
    </Trace>  
  </Traces>  
  
<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405788</Blueprin  
tUrl>  
  <Direction>To</Direction>  
  <IsSuspect>>false</IsSuspect>  
  <Label>4405788: Traveler Booking</Label>  
  
<Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/44  
05788</Link>  
  <ProjectId>4405707</ProjectId>  
  <Type>Reuse</Type>  
</Trace>  
<Trace>  
  <ArtifactId>4405707</ArtifactId>  
  <ArtifactPropertyName>BP Airways 3.0</ArtifactPropertyName>  
  <BlueprintUrl>http://today?ArtifactId=4405707</BlueprintUrl>  
  <Direction>From</Direction>  
  <IsSuspect>>false</IsSuspect>  
  <Label>4405707: BP Airways 3.0</Label>  
  
<Link>http://production.blueprintcloud.com/api/v1/projects/4405707</Link>  
  <ProjectId>4405707</ProjectId>  
  <Type>Parent</Type>  
</Trace>  
</Traces>
```

```

    <Version>5</Version>
  </Artifact>
  <Artifact>
    <AreAttachmentsReadOnly>true</AreAttachmentsReadOnly>
    <AreDocumentReferencesReadOnly>true</AreDocumentReferencesReadOnly>
    <ArtifactTypeId>317881</ArtifactTypeId>
    <ArtifactTypeName>Business Process Diagram</ArtifactTypeName>
    <Attachments>
      <Attachment>
        <FileName>Contract_of_Carriage.pdf</FileName>
        <Id>103545</Id>
        <IsReadOnly>true</IsReadOnly>
      </Attachment>
    </Attachments>
    <Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405788/attachments/103545</Link>
  </Artifact>
  <BaseArtifactType>BusinessProcess</BaseArtifactType>

  <BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405788</BlueprintUrl>
  <Id>4405788</Id>
  <Name>Traveler Booking</Name>
  <ParentId>4405707</ParentId>
  <ProjectId>4405707</ProjectId>
  <Traces>
    <Trace>
      <ArtifactId>4405717</ArtifactId>
      <ArtifactPropertyName>Traveler Booking</ArtifactPropertyName>
    </Trace>
  </Traces>
  <BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405717</BlueprintUrl>
  <Direction>From</Direction>
  <IsSuspect>>false</IsSuspect>
  <Label>4405717: Traveler Booking</Label>

  <Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405717</Link>
  <ProjectId>4405707</ProjectId>
  <Type>Reuse</Type>
</Trace>
<Trace>
  <ArtifactId>4405707</ArtifactId>
  <ArtifactPropertyName>BP Airways 3.0</ArtifactPropertyName>
</Trace>
<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405707</BlueprintUrl>
<Direction>From</Direction>
<IsSuspect>>false</IsSuspect>
<Label>4405707: BP Airways 3.0</Label>

<Link>http://production.blueprintcloud.com/api/v1/projects/4405707</Link>
<ProjectId>4405707</ProjectId>
<Type>Parent</Type>
</Trace>
</Traces>
<Version>1</Version>
</Artifact>
</Artifacts>

```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/artifacts?traces=all&attachments=true
```

Request Body

```
filter={artifact-type eq 'Business Process Diagram';}
```

Response Header

```
{'content-length': '2008', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 10:47:33 GMT', 'expires': '-1', 'content-type': 'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[
  {
    "Id": 4405717,
    "Name": "Traveler Booking",
    "ProjectId": 4405707,
    "Version": 5,
    "ParentId": 4405707,
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405717",
    "ArtifactTypeId": 317881,
    "ArtifactTypeName": "Business Process Diagram",
    "BaseArtifactType": "BusinessProcess",
    "Traces": [
      {
        "Type": "Reuse",
        "Direction": "To",
        "ProjectId": 4405707,
        "ArtifactId": 4405788,
        "ArtifactPropertyName": "Traveler Booking",
        "Label": "4405788: Traveler Booking",
        "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405788",
        "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405788",
        "IsSuspect": false
      },
      {
        "Type": "Parent",
        "Direction": "From",
        "ProjectId": 4405707,
        "ArtifactId": 4405707,
        "ArtifactPropertyName": "BP Airways 3.0",

```

```
        "Label": "4405707: BP Airways 3.0",
        "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405707",
        "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707",
        "IsSuspect": false
    }
],
"Attachments": [
    {
        "Id": 103544,
        "FileName": "Contract_of_Carriage.pdf",
        "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405717
/attachments/103544"
    }
]
},
{
    "Id": 4405788,
    "Name": "Traveler Booking",
    "ProjectId": 4405707,
    "Version": 1,
    "ParentId": 4405707,
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405788",
    "ArtifactTypeId": 317881,
    "ArtifactTypeName": "Business Process Diagram",
    "BaseArtifactType": "BusinessProcess",
    "AreAttachmentsReadOnly": true,
    "AreDocumentReferencesReadOnly": true,
    "Traces": [
        {
            "Type": "Reuse",
            "Direction": "From",
            "ProjectId": 4405707,
            "ArtifactId": 4405717,
            "ArtifactPropertyName": "Traveler Booking",
            "Label": "4405717: Traveler Booking",
            "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405717",
            "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/440571
7",
            "IsSuspect": false
        },
        {
            "Type": "Parent",
            "Direction": "From",
            "ProjectId": 4405707,
            "ArtifactId": 4405707,
            "ArtifactPropertyName": "BP Airways 3.0",
            "Label": "4405707: BP Airways 3.0",
            "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405707",
            "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707",
            "IsSuspect": false
        }
    ]
},
],
```

```
    "Attachments": [  
      {  
        "Id": 103545,  
        "FileName": "Contract_of_Carriage.pdf",  
        "Link":  
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405788  
/attachments/103545",  
        "IsReadOnly": true  
      }  
    ]  
  }  
]
```

Get Artifact

```
/api/v1/projects/{projectId}/artifacts/{artifactId}?status={status}&comments={comments}&traces={traces}&attachments={attachments}&richtextasplain={richtextasplain}&inlinecss={inlinecss}&content={content}
```

DESCRIPTION Retrieves a single artifact by Project ID and Artifact ID and returns information about the artifact.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/{ArtifactId}

- **ProjectId**: The ID of the project that contains the artifact with the attachment you want to retrieve.
- **ArtifactId**: The ID of the artifact containing the attachment you want to retrieve.

SUPPORTED METHODS

- **HEAD**
- **GET**

REQUEST URI PARAMETERS

- **Status (optional)**: Defines whether or not the status of the artifact should be loaded. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **Comments (optional)**: Indicates whether to retrieve comments of the artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **Traces (optional)**: Indicates whether to retrieve traces of the artifact. Valid options include: **None**, **All**, **Parent**, **Child**, **Manual**, **Reuse** and **Other**. The default is **None** if not specified. The default is **All** if the parameter is included in the URI with no given value.
- **Attachments (optional)**: Indicates whether to retrieve information about the attachments of the artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- **RichTextAsPlain (optional)**: Defines whether or not to retrieve all rich-text properties as Plain Text instead of HTML. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

- **InlineCSS (optional)**: Defines whether or not to retrieve all rich-text properties with locally defined or inline styles. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value. When this parameter is set to **false**, rich-text properties return internal styles that are defined within the `<head>` section of the HTML.

Note: This parameter is supported in Blueprint 6.1 and later.

- **Content (optional)**: Defines whether or not to retrieve the artifact's content. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715?comments=true&traces=all&properties=true&attachments=true
```

Response Header

```
{'content-length': '4669', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Tue, 07 May 2013 16:56:47 GMT', 'expires': '-1', 'content-type': 'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>
<Artifact xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <AreAttachmentsReadOnly>true</AreAttachmentsReadOnly>
  <AreDocumentReferencesReadOnly>true</AreDocumentReferencesReadOnly>
  <AreTracesReadOnly>true</AreTracesReadOnly>
  <ArtifactTypeId>317929</ArtifactTypeId>
  <ArtifactTypeName>ReUse_TReq</ArtifactTypeName>
  <Attachments>
    <Attachment>
```



```

        <FileName>EditReservations.txt</FileName>
        <Id>103542</Id>
        <IsReadOnly>>true</IsReadOnly>

<Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/44
05715/attachments/103542</Link>
    </Attachment>
</Attachments>
<BaseArtifactType>TextualRequirement</BaseArtifactType>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405715</Blueprin
tUrl>
    <Comments>
        <Comment>
            <Author>
                <DisplayName>Default Instance Admin</DisplayName>
                <Id>1</Id>
            </Author>
            <Description>&lt;html&gt;&lt;head&gt;&lt;style
type="text/css"&gt;.c0 { padding: 1px 0px 0px; font-size: 11px } .c1 { font-
size: 11px } .c2 { margin: 0px; font-size: 11px }
&lt;/style&gt;&lt;/head&gt;&lt;body class="c0"&gt;&lt;div class="c1"&gt;&lt;p
class="c2"&gt;&lt;span class="c1"&gt;Please
review.&lt;/span&gt;&lt;/p&gt;&lt;/div&gt;&lt;/body&gt;&lt;/html&gt;</Descripti
on>
                <Id>53978</Id>
                <LastModified>2014-12-22T19:49:31.8270000Z</LastModified>
                <Version>2</Version>
                <IsClosed>>false</IsClosed>
                <Status>Open</Status>
            </Comment>
        </Comments>
    <Id>4405715</Id>
    <Name>Edit Reservations</Name>
    <ParentId>4405707</ParentId>
    <ProjectId>4405707</ProjectId>
    <Properties>
        <Property>
            <BasePropertyType>Text</BasePropertyType>
            <IsReadOnly>>true</IsReadOnly>
            <IsRichText>>false</IsRichText>
            <Name>ID</Name>
            <PropertyTypeId>304155</PropertyTypeId>
            <TextOrChoiceValue>TR_RU4405715</TextOrChoiceValue>
        </Property>
        <Property>
            <BasePropertyType>Text</BasePropertyType>
            <IsRichText>>false</IsRichText>
            <Name>Name</Name>
            <PropertyTypeId>304156</PropertyTypeId>
            <TextOrChoiceValue>Edit Reservations</TextOrChoiceValue>
        </Property>
        <Property>
            <BasePropertyType>Text</BasePropertyType>
            <IsRichText>>true</IsRichText>
            <Name>Description</Name>
            <PropertyTypeId>304157</PropertyTypeId>
            <TextOrChoiceValue>&lt;html&gt;&lt;head&gt;&lt;style
type="text/css"&gt;.c0 { padding: 1px 0px 0px; font-size: 10.6700000762939px }
.c1 { font-size: 10.6700000762939px } .c2 { margin: 0px; font-size:

```

```
10.6700000762939px } .c3 { font-size: 10.6700000762939px; line-height:
1.45000004768372 } &lt;/style&gt;&lt;/head&gt;&lt;body class="c0"&gt;&lt;div
class="c1"&gt;&lt;p class="c2"&gt;&lt;span class="c3"&gt;&lt;span
class="c1"&gt;BP Air needs to allow customers to edit their reservations
online.&lt;/span&gt;&lt;/span&gt;&lt;/p&gt;&lt;/div&gt;&lt;/body&gt;&lt;/html&
t;</TextOrChoiceValue>
  </Property>
  <Property>
    <BasePropertyType>User</BasePropertyType>
    <Name>ApprovedBy</Name>
    <PropertyTypeId>304320</PropertyTypeId>
    <UsersAndGroups />
  </Property>
  <Property>
    <BasePropertyType>Choice</BasePropertyType>
    <Choices />
    <Name>ItemType</Name>
    <PropertyTypeId>304323</PropertyTypeId>
  </Property>
  <Property>
    <BasePropertyType>User</BasePropertyType>
    <IsReadOnly>true</IsReadOnly>
    <Name>Created By</Name>
    <PropertyTypeId>304195</PropertyTypeId>
    <UsersAndGroups>
      <UserGroup i:type="User">
        <DisplayName>Default Instance Admin</DisplayName>
        <Id>1</Id>
      </UserGroup>
    </UsersAndGroups>
  </Property>
  <Property>
    <BasePropertyType>Date</BasePropertyType>
    <DateValue>2014-12-22T19:39:30.0500000Z</DateValue>
    <IsReadOnly>true</IsReadOnly>
    <Name>Created On</Name>
    <PropertyTypeId>304196</PropertyTypeId>
  </Property>
  <Property>
    <BasePropertyType>User</BasePropertyType>
    <IsReadOnly>true</IsReadOnly>
    <Name>Last Edited By</Name>
    <PropertyTypeId>304197</PropertyTypeId>
    <UsersAndGroups>
      <UserGroup i:type="User">
        <DisplayName>Default Instance Admin</DisplayName>
        <Id>1</Id>
      </UserGroup>
    </UsersAndGroups>
  </Property>
  <Property>
    <BasePropertyType>Date</BasePropertyType>
    <DateValue>2014-12-22T19:40:01.7970000Z</DateValue>
    <IsReadOnly>true</IsReadOnly>
    <Name>Last Edited On</Name>
    <PropertyTypeId>304198</PropertyTypeId>
  </Property>
</Properties>
<Traces>
  <Trace>
```

```
<ArtifactId>4405707</ArtifactId>
<ArtifactPropertyName>BP Airways 3.0</ArtifactPropertyName>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405707</BlueprintUrl>

  <Direction>From</Direction>
  <IsSuspect>>false</IsSuspect>
  <Label>4405707: BP Airways 3.0</Label>

<Link>http://production.blueprintcloud.com/api/v1/projects/4405707</Link>
  <ProjectId>4405707</ProjectId>
  <Type>Parent</Type>
</Trace>
<Trace>
  <ArtifactId>4405714</ArtifactId>
  <ArtifactPropertyName>Reserve Online</ArtifactPropertyName>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405714</BlueprintUrl>

  <Direction>From</Direction>
  <IsSuspect>>true</IsSuspect>
  <Label>4405714: Reserve Online</Label>

<Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405714</Link>
  <ProjectId>4405707</ProjectId>
  <Type>Reuse</Type>
</Trace>
</Traces>
<Version>2</Version>
</Artifact>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715?comments=true&traces=all&properties=true&attachments=true
```

Response Header

```
{'content-length': '3180', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Tue, 07 May 2013 16:52:58 GMT', 'expires': '-1', 'content-type': 'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
{
  "Id": 4405715,
  "Name": "Edit Reservations",
  "ProjectId": 4405707,
  "Version": 2,
  "ParentId": 4405707,
```

```
"BlueprintUrl": "http://production.blueprintcloud.com?ArtifactId=4405715",
"ArtifactTypeId": 317929,
"ArtifactTypeName": "ReUse_TReq",
"BaseArtifactType": "TextualRequirement",
"AreTracesReadOnly": true,
"AreAttachmentsReadOnly": true,
"AreDocumentReferencesReadOnly": true,
"Properties": [
  {
    "PropertyTypeId": 304155,
    "Name": "ID",
    "BasePropertyType": "Text",
    "TextOrChoiceValue": "TR_RU4405715",
    "IsRichText": false,
    "IsReadOnly": true
  },
  {
    "PropertyTypeId": 304156,
    "Name": "Name",
    "BasePropertyType": "Text",
    "TextOrChoiceValue": "Edit Reservations",
    "IsRichText": false
  },
  {
    "PropertyTypeId": 304157,
    "Name": "Description",
    "BasePropertyType": "Text",
    "TextOrChoiceValue": "<html><head><style type=\"text/css\">.c0 {
padding: 1px 0px 0px; font-size: 10.6700000762939px } .c1 { font-size:
10.6700000762939px } .c2 { margin: 0px; font-size: 10.6700000762939px } .c3 {
font-size: 10.6700000762939px; line-height: 1.45000004768372 }
</style></head><body class=\"c0\"><div class=\"c1\"><p class=\"c2\"><span
class=\"c3\"><span class=\"c1\">BP Air needs to allow customers to edit their
reservations online.</span></span></p></div></body></html>",
    "IsRichText": true
  },
  {
    "PropertyTypeId": 304320,
    "Name": "ApprovedBy",
    "BasePropertyType": "User",
    "UsersAndGroups": []
  },
  {
    "PropertyTypeId": 304323,
    "Name": "ItemType",
    "BasePropertyType": "Choice",
    "Choices": []
  },
  {
    "PropertyTypeId": 304195,
    "Name": "Created By",
    "BasePropertyType": "User",
    "UsersAndGroups": [
      {
        "Type": "User",
        "Id": 1,
        "DisplayName": "Default Instance Admin"
      }
    ],
    "IsReadOnly": true
  }
]
```

```

    },
    {
      "DateValue": "2014-12-22T19:39:30.0500000Z",
      "PropertyTypeId": 304196,
      "Name": "Created On",
      "BasePropertyType": "Date",
      "IsReadOnly": true
    },
    {
      "PropertyTypeId": 304197,
      "Name": "Last Edited By",
      "BasePropertyType": "User",
      "UsersAndGroups": [
        {
          "Type": "User",
          "Id": 1,
          "DisplayName": "Default Instance Admin"
        }
      ],
      "IsReadOnly": true
    },
    {
      "DateValue": "2014-12-22T19:40:01.7970000Z",
      "PropertyTypeId": 304198,
      "Name": "Last Edited On",
      "BasePropertyType": "Date",
      "IsReadOnly": true
    }
  ],
  "Comments": [
    {
      "LastModified": "2014-12-22T19:49:31.8270000Z",
      "IsClosed": false,
      "Status": "Open",
      "Id": 53978,
      "Author": {
        "Type": "User",
        "Id": 1,
        "DisplayName": "Default Instance Admin"
      },
      "Version": 2,
      "Description": "<html><head><style type=\"text/css\">.c0 { padding: 1px 0px 0px; font-size: 11px } .c1 { font-size: 11px } .c2 { margin: 0px; font-size: 11px } </style></head><body class=\"c0\"><div class=\"c1\"><p class=\"c2\"><span class=\"c1\">Please review.</span></p></div></body></html>"
    }
  ],
  "Traces": [
    {
      "Type": "Parent",
      "Direction": "From",
      "ProjectId": 4405707,
      "ArtifactId": 4405707,
      "ArtifactPropertyName": "BP Airways 3.0",
      "Label": "4405707: BP Airways 3.0",
      "BlueprintUrl":
        "http://production.blueprintcloud.com?ArtifactId=4405707",
      "Link":
        "http://production.blueprintcloud.com/api/v1/projects/4405707",
      "IsSuspect": false
    }
  ]
}

```

```
    },
    {
      "Type": "Reuse",
      "Direction": "From",
      "ProjectId": 4405707,
      "ArtifactId": 4405714,
      "ArtifactPropertyName": "Reserve Online",
      "Label": "4405714: Reserve Online",
      "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405714",
      "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405714",
      "IsSuspect": true
    }
  ],
  "Attachments": [
    {
      "Id": 103542,
      "FileName": "EditReservations.txt",
      "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715/attachments/103542",
      "IsReadOnly": true
    }
  ]
}
```

Get Child Artifacts Of Artifact

```
/api/v1/projects/{projectId}/artifacts/{artifactId}/children?properties={properties}&status={status}&comments={comments}&traces={traces}&attachments={attachments}&richtextasplain={richtextasplain}&inlineness={inlineness}&content={content}&offset={offset}&limit={limit}
```

DESCRIPTION	<p>Retrieves the child artifact(s) of the specified artifact.</p> <p>The response only includes direct children of the specified artifact. It does not include all descendants, such as grandchildren.</p> <p>Artifacts are only included in the response if the user (that is was authenticated to obtain the token) has view permissions to those artifacts.</p>
URL STRUCTURE	<p>[Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/{ArtifactId}/children</p> <ul style="list-style-type: none">■ ProjectId: The ID of the project that contains the artifact with the attachment you want to retrieve.■ ArtifactId: The ID of the artifact containing the attachment you want to retrieve.
SUPPORTED METHODS	<ul style="list-style-type: none">■ HEAD■ GET
REQUEST URI PARAMETERS	<ul style="list-style-type: none">■ Properties (optional): Defines whether the properties should be loaded for the artifacts. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ Status (optional): Defines whether or not the status of the artifact should be loaded. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ Comments (optional): Indicates whether to retrieve comments of the artifact. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ Traces (optional): Indicates whether to retrieve traces of the artifact. Valid options include: None, All, Parent, Child, Manual, Reuse and Other. The default is None if not specified. The default is All if the parameter is included in the URI with no given value.

- `Attachments` (optional): Indicates whether to retrieve information about the attachments of the artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- `RichTextAsPlain` (optional): Defines whether or not to retrieve all rich-text properties as Plain Text instead of HTML. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- `InlineCSS` (optional): Defines whether or not to retrieve all rich-text properties with locally defined or inline styles. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value. When this parameter is set to **false**, rich-text properties return internal styles that are defined within the `<head>` section of the HTML.

Note: This parameter is supported in Blueprint 6.1 and later.

- `Offset` (optional): Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- `Limit` (optional): Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.
- `Content` (optional): Defines whether or not to retrieve the artifact's content. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715/children?comments=true&traces=all&properties=true&attachments=true
```

Response Header


```
{'content-length': '4101', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 14:58:59 GMT', 'expires': '-1', 'content-type':  
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>  
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <Artifact>  
    <AreAttachmentsReadOnly>true</AreAttachmentsReadOnly>  
    <AreDocumentReferencesReadOnly>true</AreDocumentReferencesReadOnly>  
    <AreTracesReadOnly>true</AreTracesReadOnly>  
    <ArtifactTypeId>317929</ArtifactTypeId>  
    <ArtifactTypeName>ReUse_TReq</ArtifactTypeName>  
    <Attachments>  
      <Attachment>  
        <FileName>EditReservations.txt</FileName>  
        <Id>103543</Id>  
        <IsReadOnly>true</IsReadOnly>  
      </Attachment>  
    </Attachments>  
    <Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/44  
05716/attachments/103543</Link>  
    </BaseArtifactType>TextualRequirement</BaseArtifactType>  
  </Artifact>  
</Artifacts>  
<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405716</Blueprin  
tUrl>  
  <Comments />  
  <Id>4405716</Id>  
  <Name>Cancel Reservations</Name>  
  <ParentId>4405715</ParentId>  
  <ProjectId>4405707</ProjectId>  
  <Properties>  
    <Property>  
      <BasePropertyType>Text</BasePropertyType>  
      <IsReadOnly>true</IsReadOnly>  
      <IsRichText>>false</IsRichText>  
      <Name>ID</Name>  
      <PropertyTypeId>304155</PropertyTypeId>  
      <TextOrChoiceValue>TR_RU4405716</TextOrChoiceValue>  
    </Property>  
    <Property>  
      <BasePropertyType>Text</BasePropertyType>  
      <IsRichText>>false</IsRichText>  
      <Name>Name</Name>  
      <PropertyTypeId>304156</PropertyTypeId>  
      <TextOrChoiceValue>Cancel Reservations</TextOrChoiceValue>  
    </Property>  
    <Property>  
      <BasePropertyType>Text</BasePropertyType>  
      <IsRichText>true</IsRichText>  
      <Name>Description</Name>  
      <PropertyTypeId>304157</PropertyTypeId>  
      <TextOrChoiceValue>&lt;html&gt;&lt;head&gt;&lt;style  
type="text/css"&gt;&lt;.c0 { padding: 1px 0px 0px; font-size: 10.6700000762939px }  
.c1 { font-size: 10.6700000762939px } .c2 { margin: 0px; font-size:  
10.6700000762939px } .c3 { font-size: 10.6700000762939px; line-height:
```

```
1.45000004768372 } &lt;/style&gt;&lt;/head&gt;&lt;body class="c0"&gt;&lt;div
class="c1"&gt;&lt;p class="c2"&gt;&lt;span class="c3"&gt;&lt;span
class="c1"&gt;&lt;BP Air needs to allow customers to cancel their reservations
online.&lt;/span&gt;&lt;/span&gt;&lt;/p&gt;&lt;/div&gt;&lt;/body&gt;&lt;/html&g
t;&lt;/TextOrChoiceValue>
  &lt;/Property>
  &lt;Property>
    &lt;BasePropertyType>User&lt;/BasePropertyType>
    &lt;Name>ApprovedBy&lt;/Name>
    &lt;PropertyTypeId>304320&lt;/PropertyTypeId>
    &lt;UsersAndGroups />
  &lt;/Property>
  &lt;Property>
    &lt;BasePropertyType>Choice&lt;/BasePropertyType>
    &lt;Choices />
    &lt;Name>ItemType&lt;/Name>
    &lt;PropertyTypeId>304323&lt;/PropertyTypeId>
  &lt;/Property>
  &lt;Property>
    &lt;BasePropertyType>User&lt;/BasePropertyType>
    &lt;IsReadOnly>true&lt;/IsReadOnly>
    &lt;Name>Created By&lt;/Name>
    &lt;PropertyTypeId>304195&lt;/PropertyTypeId>
    &lt;UsersAndGroups>
      &lt;UserGroup i:type="User">
        &lt;DisplayName>Greg Curtis&lt;/DisplayName>
        &lt;Id>572&lt;/Id>
      &lt;/UserGroup>
    &lt;/UsersAndGroups>
  &lt;/Property>
  &lt;Property>
    &lt;BasePropertyType>Date&lt;/BasePropertyType>
    &lt;DateValue>2014-12-22T20:39:50.5100000Z&lt;/DateValue>
    &lt;IsReadOnly>true&lt;/IsReadOnly>
    &lt;Name>Created On&lt;/Name>
    &lt;PropertyTypeId>304196&lt;/PropertyTypeId>
  &lt;/Property>
  &lt;Property>
    &lt;BasePropertyType>User&lt;/BasePropertyType>
    &lt;IsReadOnly>true&lt;/IsReadOnly>
    &lt;Name>Last Edited By&lt;/Name>
    &lt;PropertyTypeId>304197&lt;/PropertyTypeId>
    &lt;UsersAndGroups>
      &lt;UserGroup i:type="User">
        &lt;DisplayName>Greg Curtis&lt;/DisplayName>
        &lt;Id>572&lt;/Id>
      &lt;/UserGroup>
    &lt;/UsersAndGroups>
  &lt;/Property>
  &lt;Property>
    &lt;BasePropertyType>Date&lt;/BasePropertyType>
    &lt;DateValue>2014-12-22T20:40:09.1400000Z&lt;/DateValue>
    &lt;IsReadOnly>true&lt;/IsReadOnly>
    &lt;Name>Last Edited On&lt;/Name>
    &lt;PropertyTypeId>304198&lt;/PropertyTypeId>
  &lt;/Property>
&lt;/Properties>
&lt;Traces>
  &lt;Trace>
    &lt;ArtifactId>4405715&lt;/ArtifactId>
```

```
        <ArtifactPropertyName>Edit  Reservations</ArtifactPropertyName>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405715</BlueprintUrl>
        <Direction>From</Direction>
        <IsSuspect>>false</IsSuspect>
        <Label>4405715:  Edit  Reservations</Label>

<Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715</Link>
        <ProjectId>4405707</ProjectId>
        <Type>Parent</Type>
    </Trace>
    <Trace>
        <ArtifactId>4405715</ArtifactId>
        <ArtifactPropertyName>Edit  Reservations</ArtifactPropertyName>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405715</BlueprintUrl>
        <Direction>From</Direction>
        <IsSuspect>>true</IsSuspect>
        <Label>4405715:  Edit  Reservations</Label>

<Link>http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715</Link>
        <ProjectId>4405707</ProjectId>
        <Type>Reuse</Type>
    </Trace>
</Traces>
    <Version>2</Version>
</Artifact>
</Artifacts>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715/children?comments=true&traces=all&properties=true&attachments=true
```

Response Header

```
{'content-length': '2743', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 10:47:37 GMT', 'expires': '-1', 'content-type': 'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[
  {
    "Id": 4405716,
    "Name": "Cancel Reservations",
    "ProjectId": 4405707,
```

```

    "Version": 2,
    "ParentId": 4405715,
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405716",
    "ArtifactTypeId": 317929,
    "ArtifactTypeName": "ReUse_TReq",
    "BaseArtifactType": "TextualRequirement",
    "AreTracesReadOnly": true,
    "AreAttachmentsReadOnly": true,
    "AreDocumentReferencesReadOnly": true,
    "Properties": [
      {
        "PropertyTypeId": 304155,
        "Name": "ID",
        "BasePropertyType": "Text",
        "TextOrChoiceValue": "TR_RU4405716",
        "IsRichText": false,
        "IsReadOnly": true
      },
      {
        "PropertyTypeId": 304156,
        "Name": "Name",
        "BasePropertyType": "Text",
        "TextOrChoiceValue": "Cancel Reservations",
        "IsRichText": false
      },
      {
        "PropertyTypeId": 304157,
        "Name": "Description",
        "BasePropertyType": "Text",
        "TextOrChoiceValue": "<html><head><style type=\"text/css\">.c0
{ padding: 1px 0px 0px; font-size: 10.6700000762939px } .c1 { font-size:
10.6700000762939px } .c2 { margin: 0px; font-size: 10.6700000762939px } .c3 {
font-size: 10.6700000762939px; line-height: 1.45000004768372 }
</style></head><body class=\"c0\"><div class=\"c1\"><p class=\"c2\"><span
class=\"c3\"><span class=\"c1\">BP Air needs to allow customers to cancel their
reservations online.</span></span></p></div></body></html>",
        "IsRichText": true
      },
      {
        "PropertyTypeId": 304320,
        "Name": "ApprovedBy",
        "BasePropertyType": "User",
        "UsersAndGroups": []
      },
      {
        "PropertyTypeId": 304323,
        "Name": "ItemType",
        "BasePropertyType": "Choice",
        "Choices": []
      },
      {
        "PropertyTypeId": 304195,
        "Name": "Created By",
        "BasePropertyType": "User",
        "UsersAndGroups": [
          {
            "Type": "User",
            "Id": 572,
            "DisplayName": "Greg Curtis"
          }
        ]
      }
    ]
  
```

```
    }
  ],
  "IsReadOnly": true
},
{
  "DateValue": "2014-12-22T20:39:50.5100000Z",
  "PropertyTypeId": 304196,
  "Name": "Created On",
  "BasePropertyType": "Date",
  "IsReadOnly": true
},
{
  "PropertyTypeId": 304197,
  "Name": "Last Edited By",
  "BasePropertyType": "User",
  "UsersAndGroups": [
    {
      "Type": "User",
      "Id": 572,
      "DisplayName": "Greg Curtis"
    }
  ],
  "IsReadOnly": true
},
{
  "DateValue": "2014-12-22T20:40:09.1400000Z",
  "PropertyTypeId": 304198,
  "Name": "Last Edited On",
  "BasePropertyType": "Date",
  "IsReadOnly": true
}
],
"Comments": [],
"Traces": [
  {
    "Type": "Parent",
    "Direction": "From",
    "ProjectId": 4405707,
    "ArtifactId": 4405715,
    "ArtifactPropertyName": "Edit Reservations",
    "Label": "4405715: Edit Reservations",
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405715",
    "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715",
    "IsSuspect": false
  },
  {
    "Type": "Reuse",
    "Direction": "From",
    "ProjectId": 4405707,
    "ArtifactId": 4405715,
    "ArtifactPropertyName": "Edit Reservations",
    "Label": "4405715: Edit Reservations",
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405715",
    "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715",
  }
]
```

```
        "IsSuspect": true
      }
    ],
    "Attachments": [
      {
        "Id": 103543,
        "FileName": "EditReservations.txt",
        "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405716
/attachments/103543",
        "IsReadOnly": true
      }
    ]
  }
]
```

Get Root Artifacts Of Project

```
/api/v1/projects/{projectId}/children?properties={properties}&status={status}&comments={comments}&traces={traces}&attachments={attachments}&richtextasplain={richtextasplain}&inlinecss={inlinecss}&offset={offset}&limit={limit}&content={content}
```

DESCRIPTION	<p>Retrieves the artifact(s) at the root level of the project.</p> <p>The response only includes artifacts at the root level of the specified project. It does not include all artifacts in the entire project.</p> <p>Artifacts are only included in the response if the user (that is was authenticated to obtain the token) has view permissions to those artifacts.</p>
URL STRUCTURE	<p>[Blueprint_URI]/api/v1/projects/{ProjectId}/children</p> <ul style="list-style-type: none">■ ProjectId: The ID of the project.
SUPPORTED METHODS	<ul style="list-style-type: none">■ HEAD■ GET
REQUEST URI PARAMETERS	<ul style="list-style-type: none">■ Properties (optional): Defines whether the properties should be loaded for the artifacts. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ Status (optional): Defines whether or not the status of the artifact should be loaded. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ Comments (optional): Indicates whether to retrieve comments of the artifact. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ Traces (optional): Indicates whether to retrieve traces of the artifact. Valid options include: None, All, Parent, Child, Manual, Reuse and Other. The default is None if not specified. The default is All if the parameter is included in the URI with no given value.■ Attachments (optional): Indicates whether to retrieve information about the attachments of the artifact. This parameter can be set to true or false. The default is false if not specified. The default is true if the parameter is included in the URI with no given value.■ RichTextAsPlain (optional): Defines whether or not to retrieve all rich-text

properties as Plain Text instead of HTML. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

- **InlineCSS (optional)**: Defines whether or not to retrieve all rich-text properties with locally defined or inline styles. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value. When this parameter is set to **false**, rich-text properties return internal styles that are defined within the `<head>` section of the HTML.

Note: This parameter is supported in Blueprint 6.1 and later.

- **Offset (optional)**: Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional)**: Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

For Blueprint version 5.4, you can use the following parameter in addition to the ones listed above:

- **Content (optional)**: Defines whether or not to retrieve the artifact's content. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/children?traces=all&attachments=true&status=true
```

Response Header

```
{'content-length': '1413', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 14:58:56 GMT', 'expires': '-1', 'content-type': 'application/xml; charset=utf-8', 'server': 'Blueprint'}
```


Response Body

```
<?xml version="1.0" ?>
<Artifacts xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Artifact>
    <AreAttachmentsReadOnly>true</AreAttachmentsReadOnly>
    <AreDocumentReferencesReadOnly>true</AreDocumentReferencesReadOnly>
    <AreTracesReadOnly>true</AreTracesReadOnly>
    <ArtifactTypeId>11544</ArtifactTypeId>
    <ArtifactTypeName>Folder</ArtifactTypeName>
    <Attachments>
      <Attachment>
        <FileName>EditReservations.txt</FileName>
        <Id>103541</Id>
        <IsReadOnly>true</IsReadOnly>
      </Attachment>
    </Attachments>
    <Link>http://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220872/attachments/103541</Link>
  </Artifact>
  <BaseArtifactType>PrimitiveFolder</BaseArtifactType>
  <BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=220872</BlueprintUrl>
  <Id>220872</Id>
  <Name>Business Analysis</Name>
  <ParentId>220870</ParentId>
  <ProjectId>220870</ProjectId>
  <Traces>
    <Trace>
      <ArtifactId>4405715</ArtifactId>
      <ArtifactPropertyName>Edit Reservations</ArtifactPropertyName>
    </Trace>
  </Traces>
  <BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=4405715</BlueprintUrl>
  <Direction>To</Direction>
  <IsSuspect>true</IsSuspect>
  <Label>4405715: Edit Reservations</Label>
  <Link>http://production.blueprintcloud.com/api/v1/projects/220870/artifacts/4405715</Link>
  <ProjectId>4405707</ProjectId>
  <Type>Reuse</Type>
</Artifact>
</Traces>
<Status>
  <IsLocked>>false</IsLocked>
  <IsReadOnly>>false</IsReadOnly>
</Status>
<Version>1</Version>
</Artifact>
<Artifact>
  <ArtifactTypeId>11544</ArtifactTypeId>
  <ArtifactTypeName>Folder</ArtifactTypeName>
  <Attachments>
    <Attachment>
      <FileName>Contract_of_Carriage.pdf</FileName>
      <Id>103544</Id>
    </Attachment>
  </Attachments>
</Artifact>
```

```
<Link>http://production.blueprintcloud.com/api/v1/projects/220870/artifacts/4405717/attachments/103544</Link>
  </Attachment>
</Attachments>
<BaseArtifactType>PrimitiveFolder</BaseArtifactType>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=220873</BlueprintUrl>
  <Id>220873</Id>
  <Name>Application</Name>
  <ParentId>220870</ParentId>
  <ProjectId>220870</ProjectId>
  <Status>
    <IsLocked>>false</IsLocked>
    <IsReadOnly>>false</IsReadOnly>
  </Status>
  <Traces />
  <Version>1</Version>
</Artifact>
<Artifact>
  <ArtifactTypeId>11544</ArtifactTypeId>
  <ArtifactTypeName>Folder</ArtifactTypeName>
  <Attachments />
  <BaseArtifactType>PrimitiveFolder</BaseArtifactType>

<BlueprintUrl>http://production.blueprintcloud.com?ArtifactId=220874</BlueprintUrl>
  <Id>220874</Id>
  <Name>Document Output</Name>
  <ParentId>220870</ParentId>
  <ProjectId>220870</ProjectId>
  <Status>
    <IsLocked>>false</IsLocked>
    <IsReadOnly>>false</IsReadOnly>
  </Status>
  <Traces />
  <Version>1</Version>
</Artifact>
</Artifacts>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/4405707/children?attachments=true&traces=all&status=true
```

Response Header

```
{'content-length': '1884', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 10:47:35 GMT', 'expires': '-1', 'content-type': 'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[
  {
    "Id": 4405715,
    "Name": "Edit Reservations",
    "ProjectId": 4405707,
    "Version": 3,
    "ParentId": 4405707,
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405715",
    "ArtifactTypeId": 317929,
    "ArtifactTypeName": "ReUse_TReq",
    "BaseArtifactType": "TextualRequirement",
    "Status": {
      "IsLocked": false,
      "IsReadOnly": false
    },
    "AreTracesReadOnly": true,
    "AreAttachmentsReadOnly": true,
    "AreDocumentReferencesReadOnly": true,
    "Traces": [
      {
        "Type": "Reuse",
        "Direction": "To",
        "ProjectId": 4405707,
        "ArtifactId": 4405716,
        "ArtifactPropertyName": "Cancel Reservations",
        "Label": "4405716: Cancel Reservations",
        "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405716",
        "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405716",
        "IsSuspect": true
      }
    ],
    "Attachments": [
      {
        "Id": 103542,
        "FileName": "EditReservations.txt",
        "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405715/attachments/103542",
        "IsReadOnly": true
      }
    ]
  },
  {
    "Id": 4405717,
    "Name": "Traveler Booking",
    "ProjectId": 4405707,
    "Version": 5,
    "ParentId": 4405707,
    "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405717",
    "ArtifactTypeId": 317881,
    "ArtifactTypeName": "Standard Business Process",
    "BaseArtifactType": "BusinessProcess",
    "Status": {
```

```
    "IsLocked": false,
    "IsReadOnly": false
  },
  "Traces": [
    {
      "Type": "Reuse",
      "Direction": "To",
      "ProjectId": 4405707,
      "ArtifactId": 4405788,
      "ArtifactPropertyName": "Traveler Booking",
      "Label": "4405788: Traveler Booking",
      "BlueprintUrl":
"http://production.blueprintcloud.com?ArtifactId=4405788",
      "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405788",
      "IsSuspect": false
    },
  ],
  "Attachments": [
    {
      "Id": 103544,
      "FileName": "Contract_of_Carriage.pdf",
      "Link":
"http://production.blueprintcloud.com/api/v1/projects/4405707/artifacts/4405717/attachments/103544"
    }
  ]
}
```

Add Artifact

Note: This request is supported in Blueprint 5.4 and later.

`api/v1/projects/{projectId}/artifacts`

DESCRIPTION	Adds a single artifact to the specified project.
URL	[Blueprint_URI]/api/v1/projects/{projectId}/artifacts
STRUCTURE	<ul style="list-style-type: none">ProjectId: The ID of the project that you want to add the artifact to.
SUPPORTED METHODS	<ul style="list-style-type: none">POST
REQUEST BODY	The new artifact must be defined in the body of the request. Read more about how to define an artifact in the request body.

Blueprint result codes

Your response may contain `resultCode` and `message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- 201 (Successfully Created)
- 401 (Unauthorized)
 - The user does not have edit permissions for the artifact.
- 404 (Not Found)
 - Artifact not found in the project.
 - Artifact type not found.
- 410 (Invalid Value)

This error can occur for a variety of reasons, such as:

- Some of artifact properties have invalid values. (The error is on the artifact level if any artifact properties have errors.)
- No properties are specified to be updated.
- The property value cannot be empty.
- Custom value and choices cannot be specified simultaneously.
- Property does not support custom values.
- Only single choice is allowed.
- Specified choice value does not exist.
- Value must be less than the maximum value.

- Value must be greater than the minimum value.
- Limited count of decimal places.
- Invalid rich text property value.
- Specified user or group does not exist.
- **416** (Cannot Update Read-Only Property)
 - Read-only property cannot be changed.
 - ID Property cannot be updated.
- **417** (Invalid Artifact Property Type)
 - Unsupported artifact property type.
 - Property not found.
 - The property type has been removed.
 - Image property update is not supported.
- **420** (Incorrect Folder Structure)
 - Only the project or another folder can be the parent for the folder artifact.
- **500** (Error)
 - A system error has occurred.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/222996/artifacts
```

Request Body

```
<?xml version="1.0" ?>
<Artifact xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <ArtifactTypeId>25</ArtifactTypeId>
  <Name>New Textual Requirement</Name>
  <ParentId>10</ParentId>
  <Properties>
    <Property>
      <PropertyTypeId>235658</PropertyTypeId>
      <TextOrChoiceValue>Sample Description</TextOrChoiceValue>
    </Property>
  </Properties>
</Artifact>
```

```
<NumberValue>15</NumberValue>
  <PropertyTypeId>235659</PropertyTypeId>
</Property>
</Properties>
</Artifact>
```

Response Body

```
<?xml version="1.0" ?>
<PublishResultLog xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <PublishResultLogEntry>
    <ArtifactId>220914</ArtifactId>
    <Message>Published Successfully</Message>
    <ProjectId>220870</ProjectId>
    <ResultCode>200</ResultCode>
  </PublishResultLogEntry>
</PublishResultLog>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/222996/artifacts
```

Request Body

```
{
  "Name": "New Textual Requirement",
  "ParentId": 10,
  "ArtifactTypeId": 25,
  "Properties": [
    {
      "PropertyTypeId": 235658,
      "TextOrChoiceValue": "Sample Description"
    },
    {
      "PropertyTypeId": 235659,
      "NumberValue": 15.0
    }
  ]
}
```

Response Body

```
{
  "Artifact": {
    "Id": 14,
    "Name": "New Textual Requirement",
    "ParentId": 10,
```

```
"ArtifactTypeId": 25,
"ArtifactTypeName": "Textual Requirement",
"BaseArtifactType": "TextualRequirement",
"Status": {
  "IsLocked": false,
  "IsReadOnly": false
},
"Properties": [
  {
    "PropertyTypeId": 235696,
    "Name": "Created By",
    "BasePropertyType": "User",
    "UsersAndGroups": [
      {
        "Type": "User",
        "Id": 267,
        "DisplayName": "John Smith"
      }
    ]
  },
  {
    "DateValue": "2013-10-16T15:44:51.5120215Z",
    "PropertyTypeId": 235698,
    "Name": "Created On",
    "BasePropertyType": "Date"
  },
  {
    "PropertyTypeId": 235698,
    "Name": "Last Edited By",
    "BasePropertyType": "User",
    "UsersAndGroups": [
      {
        "Type": "User",
        "Id": 267,
        "DisplayName": "John Smith"
      }
    ]
  },
  {
    "DateValue": "2013-10-16T15:44:51.5120215Z",
    "PropertyTypeId": 235699,
    "Name": "Last Edited On",
    "BasePropertyType": "Date"
  },
  {
    "PropertyTypeId": 235658,
    "Name": "Description",
    "BasePropertyType": "Text",
    "TextOrChoiceValue": "Sample Description"
  },
  {
    "PropertyTypeId": 235659,
    "Name": "Level",
    "BasePropertyType": "Number",
    "NumberValue": 15.0
  }
]
"Message": "Success",
"ResultCode": 201
```



```
}
```

Update Artifacts

```
/api/v1/projects/{projectId}/artifacts
```

DESCRIPTION Updates properties of the specified artifact(s).
You can update any artifact property that is not read-only. Typically, **Name**, **Description** and any custom properties are not read-only and can be modified.
You cannot, however, modify data that is always read-only, such as the date the artifact was last updated.

Note: Reused artifacts may be read-only or contain read-only properties. Instance administrators can enable a setting that only allows you to modify certain properties, attachments or traces in a reused artifact.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts

- **ProjectId:** The ID of the project that contains the artifact with the attachment you want to retrieve.

SUPPORTED METHODS ■ **PATCH**

Note: You can [override the PATCH method](#) and use the [POST method](#) instead.

REQUEST BODY The list of artifacts that you want to publish must be included in the body of the request. Read more about how to specify a [list of artifacts](#) in the request body.

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - The artifact updated successfully.
- **401 (Unauthorized)**
 - The user does not have edit permissions for the artifact.
- **404 (Not Found)**
 - Artifact not found in the project.
 - Artifact type not found.

- **410 (Invalid Value)**

This error can occur for a variety of reasons, such as:

- Some of artifact properties have invalid values. (The error is on the artifact level if any artifact properties have errors.)
- No properties are specified to be updated.
- The property value cannot be empty.
- Custom value and choices cannot be specified simultaneously.
- Property does not support custom values.
- Only single choice is allowed.
- Specified choice value does not exist.
- Value must be less than the maximum value.
- Value must be greater than the minimum value.
- Limited count of decimal places.
- Invalid rich text property value.
- Specified user or group does not exist.

- **411 (Newer Revision Exists)**

- A newer artifact version exists.

- **414 (Locked by Another User)**

- Artifact is locked by another user.
- Cannot check out the artifact.

- **416 (Cannot Update Read-Only Property)**

- Read-only property cannot be changed.
- ID Property cannot be updated.

- **417 (Invalid Artifact Property Type)**

- Unsupported artifact property type.
- Property not found.
- The property type has been removed.
- Image property update is not supported.

- **418 (Specified Version Exceeds the Latest One)**

- The specified version is greater than the latest one.

- **500 (Error)**

- A system error has occurred.

- **502 (Concurrency Exception)**

- Another operation is being executed on the artifact.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/?
```

Request Body

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>220914</Id>
    <Properties>
      <Property>
        <PropertyTypeId>9988</PropertyTypeId>
        <TextOrChoiceValue>Traveler Booking 1</TextOrChoiceValue>
      </Property>
    </Properties>
  </Artifact>
</Artifacts>
```

Response Header

```
{'content-length': '282', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 14:59:02 GMT', 'expires': '-1', 'content-type':
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>
<ArtifactUpdateResults xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ArtifactUpdateResult>
    <ArtifactId>220914</ArtifactId>
    <Message>Success</Message>
    <ResultCode>200</ResultCode>
  </ArtifactUpdateResult>
</ArtifactUpdateResults>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/?
```

Request Body

```
[
  {
    "Id": 220914,
    "Properties": [
      {
        "PropertyTypeId": 9988,
        "TextOrChoiceValue": "Traveler Booking 1"
      }
    ]
  }
]
```

Response Header

```
{'content-length': '60', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 10:47:42 GMT', 'expires': '-1', 'content-type':
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[
  {
    "Message": "Success",
    "ResultCode": 200,
    "ArtifactId": 220914
  }
]
```

Delete Artifact

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}
```

DESCRIPTION Deletes the specified artifact. You can also delete all children of the specified artifact.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/{ArtifactId}

- **ProjectId:** The ID of the project that contains the artifact you want to delete.
- **ArtifactId:** The ID of the artifact you want to delete.

SUPPORTED METHODS

- **DELETE**

Note: You can [override the DELETE method](#) and use the [POST](#) method instead.

REQUEST URI PARAMETERS

- **Recursively (optional):** Specifies whether or not to also delete all child artifacts of the specified artifact. This parameter can be set to **true** or **false**. The default is **false** if not specified. If the parameter is false, the artifact can be deleted only if it does not have children. The default is **true** if the parameter is included in the URI with no given value.

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - The artifact updated successfully.
- **401 (Unauthorized)**
 - The user does not have edit permissions for the artifact.
- **414 (Locked by Another User)**
 - Artifact is locked by another user.
 - Cannot check out the artifact.
- **427 (Has Children)**
 - Artifact has children and deletion is not recursive.

- 500 (Error)
 - A system error has occurred.
- 502 (Concurrency Exception)
 - Another operation is being executed on the artifact.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914?Recurisively=True
```

Response Body

```
<?xml version="1.0" ?>
<ArtifactDeleteResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <ArtifactDeleteResult>
    <ArtifactId>1</ArtifactId>
    <Message>Success</Message>
    <ResultCode>200</ResultCode>
  </ArtifactDeleteResult>
  <ArtifactDeleteResult>
    <ArtifactId>12</ArtifactId>
    <Message>Success</Message>
    <ResultCode>200</ResultCode>
  </ArtifactDeleteResult>
</ArtifactDeleteResults>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914?Recurisively=True
```

Response Body

```
[
  {
    "ArtifactId": 1,
    "Message": "Success",
    "ResultCode": 200
  },
  {
    "ArtifactId": 12,
    "Message": "Success",
    "ResultCode": 200
  }
]
```


Publish Artifacts

```
/api/v1/vc/publish
```

DESCRIPTION Publishes changes of the specified artifact(s).

URL [Blueprint_URI]/api/v1/vc/publish
STRUCTURE

SUPPORTED METHODS ■ **POST**

REQUEST BODY The list of artifacts that you want to publish must be included in the body of the request. Read more about how to specify a [list of artifacts](#) in the request body.

REQUEST URI PARAMETERS ■ **KeepLock** (optional): Defines the whether or not to keep the lock after publishing the artifacts. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

Blueprint result codes

Your response may contain `resultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200** (OK)
 - The artifact updated successfully.
- **404** (Not Found)
 - Artifact not found in the project.
 - Artifact type not found.
- **412** (Nothing to Publish)
 - The artifact does not have any changes to publish.
- **415** (Not All Dependent Artifacts Included)
 - This operation does not include all dependent artifacts to be published or discarded.
- **500** (Error)
 - A system error has occurred.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/vc/publish?
```

Request Body

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>220914</Id>
    <ProjectId>220870</ProjectId>
  </Artifact>
</Artifacts>
```

Response Header

```
{'content-length': '318', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 14:59:08 GMT', 'expires': '-1', 'content-type':
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>
<PublishResultLog xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <PublishResultLogEntry>
    <ArtifactId>220914</ArtifactId>
    <Message>Published Successfully</Message>
    <ProjectId>220870</ProjectId>
    <ResultCode>200</ResultCode>
  </PublishResultLogEntry>
</PublishResultLog>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/vc/publish?
```

Request Body

```
[  
  {  
    "Id": "220914",  
    "ProjectId": "220870"  
  }  
]
```

Response Header

```
{'content-length': '94', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 10:47:50 GMT', 'expires': '-1', 'content-type':  
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[  
  {  
    "ProjectId": 220870,  
    "ResultCode": 200,  
    "Message": "Published Successfully",  
    "ArtifactId": 220914  
  }  
]
```

Discard Artifacts

```
/api/v1/vc/discard
```

DESCRIPTION Discards changes to the specified artifact(s).

URL STRUCTURE [Blueprint_URI]/api/v1/vc/discard

SUPPORTED METHODS ■ **POST**

REQUEST BODY The list of artifacts that you want to publish must be included in the body of the request. Read more about how to specify a [list of artifacts](#) in the request body.

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - Successfully discarded the artifact.
- **404 (Not Found)**
 - Artifact not found in the project.
 - Artifact type not found.
- **413 (Nothing to Discard)**
 - The artifact does not have any changes to discard.
- **415 (Not All Dependent Artifacts Included)**
 - This operation does not include all dependent artifacts to be published or discarded.
- **500 (Error)**
 - A system error has occurred.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/vc/discard?
```

Request Body

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>220914</Id>
    <ProjectId>220870</ProjectId>
  </Artifact>
</Artifacts>
```

Response Header

```
{'content-length': '318', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 14:59:04 GMT', 'expires': '-1', 'content-type':
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>
<DiscardResultLog xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <DiscardResultLogEntry>
    <ArtifactId>220914</ArtifactId>
    <Message>Discarded Successfully</Message>
    <ProjectId>220870</ProjectId>
    <ResultCode>200</ResultCode>
  </DiscardResultLogEntry>
</DiscardResultLog>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/vc/discard?
```

Request Body

```
[  
  {  
    "Id": "220914",  
    "ProjectId": "220870"  
  }  
]
```

Response Header

```
{'content-length': '94', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 10:47:45 GMT', 'expires': '-1', 'content-type':  
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
[  
  {  
    "ProjectId": 220870,  
    "ResultCode": 200,  
    "Message": "Discarded Successfully",  
    "ArtifactId": 220914  
  }  
]
```

List Unpublished Artifacts

Note: This request is supported in Blueprint 5.4 and later.

`api/v1/vc/unpublishedArtifacts`

DESCRIPTION Retrieves a list of artifacts with unpublished changes. This service only returns artifacts in the list if the changes were made by the current user.

URL STRUCTURE [Blueprint_URI]/api/v1/vc/unpublishedArtifacts

SUPPORTED METHODS

- **GET**
- **HEAD**

REQUEST URI PARAMETERS

- **Offset (optional):** Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional):** Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/vc/unpublishedArtifacts?offset=3&limit=13
```

Response Body

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <ArtifactTypeId>1</ArtifactTypeId>
    <ArtifactTypeName>Actor</ArtifactTypeName>
    <BaseArtifactType>Actor</BaseArtifactType>
    <Id>1</Id>
    <Name>New Actor 1</Name>
    <ParentId>10</ParentId>
  </Artifact>
  <Artifact>
    <ArtifactTypeId>2</ArtifactTypeId>
    <ArtifactTypeName>Textual Requirement</ArtifactTypeName>
    <BaseArtifactType>TextualRequirement</BaseArtifactType>
    <Id>2</Id>
    <Name>New Textual Requirement 1</Name>
    <ParentId>10</ParentId>
  </Artifact>
</Artifacts>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/vc/unpublishedArtifacts?offset=3&limit=13
```

Response Body

```
[
  {
    "Id": 1,
    "Name": "New Actor 1",
    "ParentId": 10,
    "ArtifactTypeId": 1,
    "ArtifactTypeName": "Actor",
    "BaseArtifactType": "Actor"
  },
  {
    "Id": 2,
    "Name": "New Textual Requirement 1",
    "ParentId": 10,
    "ArtifactTypeId": 2,
    "ArtifactTypeName": "Textual Requirement",
    "BaseArtifactType": "TextualRequirement"
  }
]
```


Get Collection

```
/api/v1/projects/{projectId}/collections/{collectionId}
```

DESCRIPTION Retrieves the collection in the specified project.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/collections/{collectionId}

- **projectId**: The ID of the project that contains the collection you want to retrieve.
- **collectionId**: The ID of the collection you want to retrieve.

SUPPORTED METHODS

- **HEAD**
- **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/101/collections/131
```

Response Body

```
<?xml version="1.0" ?>
<Collection xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Id>131</Id>
  <Name>My Collection 1</Name>
  <ProjectId>101</ProjectId>
  <ContentArtifactIds>
    <Id>108</Id>
    <Id>109</Id>
    <Id>111</Id>
    <Id>120</Id>
  </ContentArtifactIds>
</Collection>
```

```
</ContentArtifactIds>  
</Collection>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/101/collections/131
```

Response Body

```
{  
  "ContentArtifactIds": [  
    108,  
    109,  
    111,  
    120  
  ],  
  "Id": 131,  
  "Name": "My Collection 1",  
  "ProjectId": 101  
}
```

List Collections

```
/api/v1/projects/{projectId}/collections
```

DESCRIPTION Lists collections in the specified project.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/collections

- **ProjectId**: The ID of the project that contains the collections you want to retrieve.

SUPPORTED METHODS

- **HEAD**
- **GET**

REQUEST URI PARAMETERS

- **Offset (optional)**: Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional)**: Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/101/collections
```

Response Body

```
<?xml version="1.0" ?>  
<Collection xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
```

```
<Id>131</Id>
<Name>My Collection 1</Name>
<ProjectId>101</ProjectId>
<ContentArtifactIds>
  <Id>108</Id>
  <Id>109</Id>
  <Id>111</Id>
  <Id>120</Id>
</ContentArtifactIds>
</Collection>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/101/collections/131
```

Response Body

```
{
  "ContentArtifactIds": [
    108,
    109,
    111,
    120
  ],
  "Id": 131,
  "Name": "My Collection 1",
  "ProjectId": 101
}
```

Get Attachments

```
/api/v1/projects/{projectId}/artifacts/{artifactId}/attachments/  
{attachmentId}
```

DESCRIPTION Retrieves attachment content.

URL [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/
STRUCTURE {ArtifactId}/attachments/{AttachmentId}

- **ProjectId**: The ID of the project that contains the artifact with the attachment you want to retrieve.
- **ArtifactId**: The ID of the artifact containing the attachment you want to retrieve.
- **AttachmentId**: Defines the ID of the attachment. You can obtain an attachment ID by setting the `Attachments` parameter to **True** in any request that retrieves artifacts (example [Get Artifact](#)).

SUPPORTED METHODS

- **HEAD**
- **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/a  
ttachments/3194?
```

Response Header

```
{'content-length': '386481', 'content-disposition': 'attachment; filename="'  
filename=Contract_of_Carriage.pdf'", 'expires': '-1', 'server': 'Blueprint',
```

```
'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Tue, 07 May 2013  
17:02:01 GMT', 'content-type': 'application/octet-stream'}
```

Response Body

The attachment data is contained in the response body.

The response header contains important information about the attachment, such as:

- `content-length`: The size of the response (that is, the attachment).
- `filename`: The name of the attachment file.
- `content-type`: The response content type, which is `octet-stream` in this case.

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/a  
ttachments/3194?
```

Response Header

```
{'content-length': '386481', 'content-disposition': 'attachment;  
filename="Contract_of_Carriage.pdf"', 'expires': '-1', 'server': 'Blueprint',  
'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Tue, 07 May 2013  
16:58:56 GMT', 'content-type': 'application/octet-stream'}
```

Response Body

The attachment data is contained in the response body.

The response header contains important information about the attachment, such as:

- `content-length`: The size of the response (that is, the attachment).
- `filename`: The name of the attachment file.
- `content-type`: The response content type, which is `octet-stream` in this case.

Add Attachment

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/attachments
```

DESCRIPTION Adds a single file attachment to the specified artifact.
If the specified artifact is a Document artifact, the file is uploaded as the document artifact file (not as an attachment in the utility panel).

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/{artifactId}/attachments

- **ProjectId:** The ID of the project containing the artifact you want to add the attachment to.
- **ArtifactId:** The ID of the artifact you want to add the attachment to.

SUPPORTED METHODS ■ **POST**

REQUEST BODY There are two ways to upload an attachment:

- To add an attachment as a part of the request body, use the `Content-MD5` header and manually calculate a MD5 token for the value.
- Instead of providing an attachment as a part of the request body, you have the option of adding different sets of data by using `MultipartContent` MIME. For `MultiPartContent`, you must specify the `Content-Type` header as `multipart/form-data` (that is, `Content-Type: multipart/form-data; boundary={token}`) and use `boundary` parameters. For more information on using `MultipartContent`, see http://www.w3.org/Protocols/rfc1341/7_2_Multipart.html. For a reference in C#, see this [example of a MultiPartContent request](#).

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/attachments
```

Request Header (Using Request Body to Add Attachment)

```
Authorization: BlueprintToken  
IzRJb4XcD400YewT/3/bclvTZ9tjRtwvNXz1J1N6/j4jzp4TpEEI4j  
Content-MD5: 86E8C9EF442FFE19F3F7502B9E11B5E4  
Content-disposition: attachment;filename="BitmapAttachment.bmp"  
Content-Type: application/octet-stream  
Content-Length: 9270
```

Response Body

```
<?xml version="1.0" ?>  
<Attachment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <FileName>BitmapAttachment.png</FileName>  
  <Id>51097</Id>  
  
  <Link>http://localhost:61928/api/v1/projects/2080006/artifacts/2080014/attachments/51097</Link>  
</Attachment>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/attachments
```

Request Header (Using Request Body to Add Attachment)

```
Authorization: BlueprintToken  
IzRJb4XcD400YewT/3/bclvTZ9tjRtwvNXz1J1N6/j4jzp4TpEEI4j  
Content-MD5: 86E8C9EF442FFE19F3F7502B9E11B5E4  
Content-disposition: attachment;filename="BitmapAttachment.bmp"  
Content-Type: application/octet-stream  
Content-Length: 9270
```


Response Body

```
{  
  "Id": 51097,  
  "FileName": "BitmapAttachment.png",  
  "Link":  
  "http://today/api/v1/projects/2080006/artifacts/2080014/attachments/51097"  
}
```

Add Attachment to Subartifact

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/subArtifacts/  
{subArtifactId}/attachments
```

DESCRIPTION Adds a single file attachment to the specified subartifact.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/
{artifactId}/subArtifacts/{subArtifactId}/attachments

- **ProjectId:** The ID of the project containing the artifact you want to add the attachment to.
- **ArtifactId:** The ID of the artifact you want to add the attachment to.
- **SubArtifactId:** The ID of the subartifact you want to add the attachment to.
Note: As of Blueprint 5.4, the API does not support the ability to retrieve a list of subartifact IDs.

SUPPORTED METHODS ■ **POST**

REQUEST BODY There are two ways to upload an attachment:

- To add an attachment as a part of the request body, use the `Content-MD5` header and manually calculate a MD5 token for the value.
- Instead of providing an attachment as a part of the request body, you have the option of adding different sets of data by using `MultipartContent` MIME. For `MultiPartContent`, you must specify the `Content-Type` header as `multipart/form-data` (that is, `Content-Type: multipart/form-data;boundary={token}`) and use `boundary` parameters. For more information on using `MultipartContent`, see http://www.w3.org/Protocols/rfc1341/7_2_Multipart.html. For a reference in C#, see this [example of a MultiPartContent request](#).

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/subArtifacts/229080/attachments
```

Request Header (Using Request Body to Add Attachment)

```
Authorization: BlueprintToken  
IzRJb4XcD400YewT/3/bclvTZ9tjRtwtvNXz1J1N6/j4jzp4TpEEI4j  
Content-MD5: 86E8C9EF442FFE19F3F7502B9E11B5E4  
Content-disposition: attachment;filename="BitmapAttachment.bmp"  
Content-Type: application/octet-stream  
Content-Length: 9270
```

Response Body

```
<?xml version="1.0" ?>  
<Attachment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <FileName>BitmapAttachment.bmp</FileName>  
  <Id>8083</Id>  
  
  <Link>http://localhost:61928/api/v1/projects/206078/artifacts/206093/attachment  
s/8083</Link>  
</Attachment>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/subArtifacts/229080/attachments
```

Request Header (Using Request Body to Add Attachment)

```
Authorization: BlueprintToken  
IzRJb4XcD400YewT/3/bclvTZ9tjRtwtvNXz1J1N6/j4jzp4TpEEI4j  
Content-MD5: 86E8C9EF442FFE19F3F7502B9E11B5E4  
Content-disposition: attachment;filename="BitmapAttachment.bmp"  
Content-Type: application/octet-stream  
Content-Length: 9270
```

Response Body

```
{
  "Id": 8082,
  "FileName": "BitmapAttachment.bmp",
  "Link":
  "http://localhost:61928/api/v1/projects/206078/artifacts/206093/attachments/8082"
}
```

Delete Attachment

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/attachments/  
{attachmentId}
```

DESCRIPTION Deletes a single file attachment from an artifact.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/
{artifactId}/attachments/{attachmentId}

- **ProjectId:** The ID of the project that contains the attachment you want to delete.
- **ArtifactId:** The ID of the artifact containing the attachment you want to delete.
- **AttachmentId:** The ID of the attachment you want to delete. You can obtain the attachmentID via the [Get Artifact](#) request.

SUPPORTED METHODS

- **DELETE**

Note: You can [override the DELETE method](#) and use the [POST](#) method instead.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/a  
ttachments/51097
```

Response Body

```
<?xml version="1.0" ?>
  <string
xmlns="http://schemas.microsoft.com/2003/10/Serialization/">"Attachment 51097
has been successfully deleted"</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/a
ttachments/51097
```

Response Body

```
"Attachment 51097 has been successfully deleted"
```

List Groups

```
/api/v1/groups?offset={offset}&limit={limit}
```

DESCRIPTION Retrieves a list of all groups in this Blueprint instance.

URL [Blueprint_URI]/api/v1/groups
STRUCTURE

SUPPORTED METHODS

- HEAD
- GET

REQUEST URI PARAMETERS

- **Offset (optional)**: Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional)**: Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/groups?Limit=5&Offset=0
```

Response Header

```
{'content-length': '372', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'groups 0-4/74', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 14:59:16 GMT', 'content-type': 'application/xml; charset=utf-8'}
```

Response Body

```
<?xml version="1.0" ?>
<Groups xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Group>
    <Id>1</Id>
    <Name>Author</Name>
  </Group>
  <Group>
    <Id>2</Id>
    <Name>Collaborator</Name>
  </Group>
  <Group>
    <Id>3</Id>
    <Name>Dept - Sales</Name>
  </Group>
  <Group>
    <Id>4</Id>
    <Name>Dept - PreSales</Name>
  </Group>
  <Group>
    <Id>5</Id>
    <Name>Dept - Services</Name>
  </Group>
</Groups>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/groups?Limit=5&Offset=0
```

Response Header

```
{'content-length': '316', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'groups 0-4/74', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 11:43:00 GMT', 'content-type': 'application/json; charset=utf-8'}
```

Response Body

```
[
  {
    "ProjectId": null,
    "Type": "Group",
    "Id": 1,
    "Name": "Author"
  },
  {
    "ProjectId": null,
```



```
"Type": "Group",
  "Id": 2,
  "Name": "Collaborator"
},
{
  "ProjectId": null,
  "Type": "Group",
  "Id": 3,
  "Name": "Dept - Sales"
},
{
  "ProjectId": null,
  "Type": "Group",
  "Id": 4,
  "Name": "Dept - PreSales"
},
{
  "ProjectId": null,
  "Type": "Group",
  "Id": 5,
  "Name": "Dept - Services"
}
]
```

Get Group

```
/api/v1/groups/{groupId}
```

DESCRIPTION Retrieves the group by its ID.

URL [Blueprint_URI]/api/v1/groups/{GroupId}

STRUCTURE

- GroupId: The ID of the group you want to get.

SUPPORTED METHODS

- HEAD
- GET

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/groups/75?
```

Response Header

```
{'content-length': '155', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 14:59:17 GMT', 'expires': '-1', 'content-type':  
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>  
<Group xmlns="http://www.blueprintsys.com/blueprint/api/v1"  
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">  
  <Id>75</Id>  
  <Name>API Authors</Name>  
</Group>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/groups/75?
```

Response Header

```
{'content-length': '62', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 11:49:45 GMT', 'expires': '-1', 'content-type':  
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
{  
  "ProjectId": null,  
  "Type": "Group",  
  "Id": 75,  
  "Name": "API Authors"  
}
```

Add Traces

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/  
{artifactId}/traces?reconcilewithtwoway={boolean}
```

DESCRIPTION Adds trace(s) to the artifact you specify.
You can only add *manual* traces (that is, a trace manually added by a user). You cannot add *parent/child* relationships, *reuse* relationships or *other* relationships.

Note: You can add more than one trace in a single request as long as the `ArtifactId` you specify in the request body has a trace relationship with the artifact in the request URI.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/
{ArtifactId}/traces

- `ProjectId`: The ID of the project that contains the artifact you want to add the trace(s) to.
- `ArtifactId`: The ID of the artifact you want add the trace(s) to.

SUPPORTED METHODS ■ **POST**

REQUEST URI PARAMETERS ■ `ReconcileWithTwoWay` (optional): Indicates how to handle the existence of an inverse trace. This parameter can be set to **true** or **false**. If set to **true**, and an inverse trace already exists, the request does not return an error; instead, the trace **Type** is set to **TwoWay**. The default is **false** if not specified.
The default is **true** if the parameter is included in the URI with no given value.

Note: This parameter is available in Blueprint 6.3 and later.

REQUEST BODY PARAMETERS The trace(s) must be included in the body of the request. Read more about how to specify [one or more traces](#) in the request body.

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- 201 (Successfully Created)
- 400 (Bad Request)
- 421 (Trace Not Manual)
 - Only valid trace value is *Manual*.
- 424 (Trace Artifact Not Found)
 - Artifact for the trace cannot be located.
- 425 (Trace Cannot Be Managed)
 - In order to manage a trace, you need to have *Trace* permissions for both traced artifacts and *Edit* permissions for at least one artifact.
- 426 (Traced Sub-Artifact Does Not Belong To Provided Artifact)
 - Sub-artifact does not belong to the provided artifact.
- 500 (Error)
 - A system error has occurred.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Examples

Example 1: successful add trace call from artifact 777001 to 888020

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/777001/traces
```

Request Body

```
<?xml version="1.0" ?>
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Trace>
    <ArtifactId>888020</ArtifactId>
    <Direction>To</Direction>
    <IsSuspect>>false</IsSuspect>
    <ProjectId>847739</ProjectId>
    <Type>Manual</Type>
  </Trace>
</Traces>
```

Response Body

```
<?xml version="1.0" ?>
<TraceAddResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <TraceAddResult>
    <ArtifactId>888020</ArtifactId>
    <Direction>To</Direction>
    <IsSuspect>>false</IsSuspect>
    <Message>Trace between 777001 and 888020 added successfully.</Message>
    <ProjectId>847739</ProjectId>
    <ResultCode>201</ResultCode>
    <Type>Manual</Type>
  </TraceAddResult>
</TraceAddResults>
```

Example 2: failed attempt to add inverse trace from artifact 888020 to 777001

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/888020/traces
```

Request Body

```
<?xml version="1.0" ?>
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Trace>
    <ArtifactId>777001</ArtifactId>
    <Direction>To</Direction>
    <IsSuspect>>false</IsSuspect>
    <ProjectId>847739</ProjectId>
    <Type>Manual</Type>
  </Trace>
</Traces>
```

Response Body

```
<?xml version="1.0" ?>
<TraceAddResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <TraceAddResult>
    <ArtifactId>777001</ArtifactId>
    <Direction>To</Direction>
    <IsSuspect>>false</IsSuspect>
    <Message>A trace with the specified artifact/subartifact already exists.
    Provided values are Project Id: 847739, Artifact Id: 777001, Subartifact Id:
    0</Message>
    <ProjectId>847739</ProjectId>
    <ResultCode>400</ResultCode>
    <Type>Manual</Type>
```

```
</TraceAddResult>  
</TraceAddResults>
```

Example 3: using the ReconcileWithTwoWay parameter to handle an existing trace relationship

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/888020/traces?reconcilewithtwoway=true
```

Request Body

```
<?xml version="1.0" ?>  
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <Trace>  
    <ArtifactId>777001</ArtifactId>  
    <Direction>To</Direction>  
    <IsSuspect>>false</IsSuspect>  
    <ProjectId>847739</ProjectId>  
    <Type>Manual</Type>  
  </Trace>  
</Traces>
```

Response Body

```
<?xml version="1.0" ?>  
<TraceAddResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <TraceAddResult>  
    <ArtifactId>777001</ArtifactId>  
    <Direction>TwoWay</Direction>  
    <IsSuspect>>false</IsSuspect>  
    <Message>Trace between 777001 and 888020 added successfully.</Message>  
    <ProjectId>847739</ProjectId>  
    <ResultCode>201</ResultCode>  
    <Type>Manual</Type>  
  </TraceAddResult>  
</TraceAddResults>
```

JSON Examples

Example 1: successful add trace call from artifact 777001 to 888020

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/777001/traces
```

Request Body

```
[
  {
    "Type": "Manual",
    "ProjectId": 123456,
    "ArtifactId": 123,
    "IsSuspect": false
  },
  {
    "Type": "Manual",
    "ProjectId": 123456,
    "ArtifactId": 456,
    "IsSuspect": false
  },
  {
    "Type": "Manual",
    "ProjectId": 789,
    "ArtifactId": 789,
    "SubArtifactId": 789,
    "IsSuspect": false
  }
]
```

Response Body

```
[
  {
    "ProjectId": 847739,
    "ArtifactId": 888020,
    "Direction": "To",
    "Type": "Manual",
    "Message": "Trace between 777001 and 888020 added successfully.",
    "ResultCode": 201,
    "IsSuspect": false
  }
]
```

Example 2: failed attempt to add inverse trace from artifact 888020 to 777001

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/888020/traces
```

Request Body

```
[
  {
    "Type": "Manual",
    "ProjectId": 847739,
```



```
"ArtifactId": 777001,  
"Direction": "To",  
"IsSuspect": false  
}  
]
```

Response Body

```
[  
  {  
    "ProjectId": 847739,  
    "ArtifactId": 777001,  
    "Direction": "To",  
    "Type": "Manual",  
    "Message": "A trace with the specified artifact/subartifact already exists.  
Provided values are Project Id: 847739, Artifact Id: 777001, Subartifact Id:  
0",  
    "ResultCode": 400,  
    "IsSuspect": false  
  }  
]
```

Example 3: using the `ReconcileWithTwoWay` parameter to handle an existing trace relationship

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/888020/traces?recon  
cilewithtwoway=true
```

Request Body

```
[  
  {  
    "Type": "Manual",  
    "ProjectId": 847739,  
    "ArtifactId": 777001,  
    "Direction": "To",  
    "IsSuspect": false  
  }  
]
```

Response Body

```
[  
  {  
    "ProjectId": 847739,  
    "ArtifactId": 777001,  
    "Direction": "TwoWay",
```

```
"Type": "Manual",  
"Message": "Trace between 777001 and 888020 added successfully.",  
"ResultCode": 201,  
"IsSuspect": false  
}  
]
```

Move Artifact

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/traces
```

DESCRIPTION In order to *move* an artifact, you must change the parent of the artifact. By changing the parent, you are effectively moving the specified artifact and its descendants to a new location in the hierarchy.

Note: Artifacts with read-only traces cannot be moved.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/{ArtifactId}/traces

- **ProjectId:** The ID of the project that contains the artifact you want to move.
- **ArtifactId:** The ID of the artifact you want to move.

SUPPORTED METHODS

- **PATCH**
- **POST**

REQUEST BODY The new parent artifact must be specified in the request body. Read more about [how to specify the parent artifact in the request body](#).

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - The artifact updated successfully.
- **400 (Bad Request)**
- **414 (Locked by Another User)**
 - Artifact is locked by another user.
 - Cannot check out the artifact.
- **428 (New Parent Is Current One)**
 - The newly specified parent is already the current parent.
- **429 (New Parent Not Found For User)**

- The newly specified parent cannot be found for the user.
- **430 (New Parent Not In Project)**
 - The newly specified parent is from another project. Moving the artifact across projects is not supported.
- **431 (No Edit Permission On New Parent)**
 - Cannot make the specified artifact the new parent. The user does not have edit permissions for the new parent.
- **432 (New Parent Is Descendent Or Itself)**
 - The newly specified parent is either a descendent of the artifact or is the same artifact ID that was specified in the URL.
- **500: Internal Server Error**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/888020/traces
```

Request Body

```
<?xml version="1.0" ?>  
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <Trace>  
    <ArtifactId>999666</ArtifactId>  
    <Type>Parent</Type>  
  </Trace>  
</Traces>
```

Response Body

```
<?xml version="1.0" ?>  
<TraceUpdateResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <TraceUpdateResult>  
    <ArtifactId>999666</ArtifactId>
```

```
<Message>The parent of the artifact has been successfully
updated.</Message>
<ResultCode>200</ResultCode>
<Type>Parent</Type>
</TraceUpdateResult>
</TraceUpdateResults>
```

JSON Example

Request URI

```
https://blueprintcloud.com/api/v1/projects/847739/artifacts/888020/traces
```

Request Body

```
[
  {
    "Type": "Parent",
    "ArtifactId": 999666
  }
]
```

Response Body

```
[
  {
    "ArtifactId": 999666,
    "Type": "Parent",
    "Message": "The parent of the artifact has been successfully updated.",
    "ResultCode": 200
  }
]
```

Delete Traces

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/  
{artifactId}/traces?reconcilewithtwoway={boolean}
```

DESCRIPTION Deletes the specified trace(s).
You can only delete *manual* traces (that is, a trace manually added by a user). You cannot delete *parent/child* relationships or *other* relationships.

Note: You can delete more than one trace in a single request as long as the `ArtifactId` you specify in the request body has a trace relationship with the artifact in the request URI.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/artifacts/
{ArtifactId}/traces

- `ProjectId`: The ID of the project that contains the first artifact in the trace relationship you want to delete. A trace relationship always includes two artifacts.
- `ArtifactId`: The ID of the first artifact in the trace relationship you want to delete. A trace relationship always includes two artifacts.

SUPPORTED METHODS

- **DELETE**

Note: You can [override the DELETE method](#) and use the [POST method](#) instead.

REQUEST URI PARAMETERS

- `ReconcileWithTwoWay` (optional): Indicates how to handle the existence of an inverse trace. This parameter can be set to **true** or **false**, and is used in conjunction with the `Direction` parameter in the request body that is set to **To** or **From**. If `reconcileWithTwoWay` is set to **true**, and a bidirectional trace exists, the trace is not deleted; instead, the trace direction opposite to that in the request body is preserved (that is, the inverse trace replaces the bidirectional trace). The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.

Note: This parameter is available in Blueprint 6.3 and later.

REQUEST BODY The trace(s) must be included in the body of the request. Read more about [how to specify one or more traces in the request body](#).

PARAMETERS

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - The artifact updated successfully.
- **421 (Trace Not Manual)**
 - Only valid trace value is *Manual*.
- **422 (Trace Not Found)**
 - The artifact does not have the specified trace.
- **425 (Trace Cannot Be Managed)**
 - In order to manage a trace, you need to have *Trace* permissions for both traced artifacts and *Edit* permissions for at least one artifact.
- **500: Internal Server Error**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Examples

Example 1

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/847739/artifacts/847741/traces
```

Request Body

```
<?xml version="1.0" ?>  
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
```

```
<Trace>
  <ArtifactId>843925</ArtifactId>
  <Type>Manual</Type>
</Trace>
</Traces>
```

Response Body

```
<TraceDeleteResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <TraceDeleteResult>
    <ArtifactId>843925</ArtifactId>
    <Direction>TwoWay</Direction>
    <Message>Trace has been successfully deleted.</Message>
    <ResultCode>200</ResultCode>
  </TraceDeleteResult>
</TraceDeleteResults>
```

Example 2: using the ReconcileWithTwoWay parameter to retain the inverse of a bidirectional trace relationship

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/847739/artifacts/843921/traces?reconcilewithtwoway=true
```

Request Body

```
<?xml version="1.0" ?>
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Trace>
    <ArtifactId>843927</ArtifactId>
    <Direction>From</Direction>
    <Type>Manual</Type>
  </Trace>
</Traces>
```

Response Body

```
<TraceDeleteResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <TraceDeleteResult>
    <ArtifactId>843927</ArtifactId>
    <Direction>TwoWay</Direction>
    <Message>Trace has been successfully deleted.</Message>
```



```
<ResultCode>200</ResultCode>
</TraceDeleteResult>
</TraceDeleteResults>
```

JSON Examples

Example 1

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/847739/artifacts/847741/traces
```

Request Body

```
[
  {
    "Type": "Manual",
    "ArtifactId": 847750
  }
]
```

Response Body

```
[
  {
    "ArtifactId": 847750,
    "Direction": "TwoWay",
    "Type": "Manual",
    "Message": "Trace has been successfully deleted.",
    "ResultCode": 200
  }
]
```

Example 2: using the `ReconcileWithTwoWay` parameter to retain the inverse of a bidirectional trace relationship

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/847739/artifacts/843921/traces?reconcilewithtwoway=true
```

Request Body

```
[
  {
    "Type": "Manual",
    "ArtifactId": 843923,
    "Direction": "To"
  }
]
```

Response Body

```
[
  {
    "ArtifactId": 843923,
    "Direction": "TwoWay",
    "Type": "Manual",
    "Message": "Trace has been successfully deleted.",
    "ResultCode": 200
  }
]
```

List Users

```
/api/v1/users?offset={offset}&limit={limit}
```

DESCRIPTION Retrieves all users in this Blueprint instance.

URL [Blueprint_URI]/api/v1/users
STRUCTURE

SUPPORTED METHODS

- HEAD
- GET

REQUEST URI PARAMETERS

- **Offset (optional):** Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- **Limit (optional):** Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users?Limit=3&Offset=13
```

Response Header

```
{'content-length': '955', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'users 13-15/122', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 14:59:21 GMT', 'content-type': 'application/xml; charset=utf-8'}
```

Response Body

```
<?xml version="1.0" ?>
<Users xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <User>
    <DisplayName>Steph Hubert</DisplayName>
    <Firstname>Steph</Firstname>
    <Groups>
      <Group>
        <Id>19</Id>
        <Name>Dept - Product Management</Name>
      </Group>
      <Group>
        <Id>20</Id>
        <Name>Release Note Collaborators</Name>
      </Group>
      <Group>
        <Id>34</Id>
        <Name>BP Gold Group</Name>
      </Group>
    </Groups>
    <Id>15</Id>
    <Lastname>Hubert</Lastname>
    <Name>blueprint\shubert</Name>
  </User>
  <User>
    <DisplayName>Matt Dodgson</DisplayName>
    <Firstname>Matthew</Firstname>
    <Groups>
      <Group>
        <Id>7</Id>
        <Name>Feature Team - Integration</Name>
      </Group>
      <Group>
        <Id>19</Id>
        <Name>Dept - Product Management</Name>
      </Group>
      <Group>
        <Id>34</Id>
        <Name>BP Gold Group</Name>
      </Group>
    </Groups>
    <Id>16</Id>
    <Lastname>Dodgson</Lastname>
    <Name>blueprint\mdodgson</Name>
  </User>
  <User>
    <DisplayName>Ari Test</DisplayName>
    <Firstname>Ari</Firstname>
    <Groups/>
    <Id>17</Id>
    <Lastname>Test</Lastname>
    <Name>aritest</Name>
  </User>
</Users>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users?Limit=3&Offset=13
```

Response Header

```
{'content-length': '822', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'users 13-15/122', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 11:59:23 GMT', 'content-type': 'application/json; charset=utf-8'}
```

Response Body

```
[
  {
    "DisplayName": "Steph Hubert",
    "Name": "blueprint\\shubert",
    "Firstname": "Steph",
    "Lastname": "Hubert",
    "Groups": [
      {
        "ProjectId": null,
        "Type": "Group",
        "Id": 19,
        "Name": "Dept - Product Management"
      },
      {
        "ProjectId": null,
        "Type": "Group",
        "Id": 20,
        "Name": "Release Note Collaborators"
      },
      {
        "ProjectId": null,
        "Type": "Group",
        "Id": 34,
        "Name": "BP Gold Group"
      }
    ],
    "Type": "User",
    "Id": 15
  },
  {
    "DisplayName": "Matt Dodgson",
    "Name": "blueprint\\mdodgson",
    "Firstname": "Matthew",
    "Lastname": "Dodgson",
    "Groups": [
      {
        "ProjectId": null,
        "Type": "Group",
        "Id": 7,

```

```
    "Name": "Feature Team - Integration"
  },
  {
    "ProjectId": null,
    "Type": "Group",
    "Id": 19,
    "Name": "Dept - Product Management"
  },
  {
    "ProjectId": null,
    "Type": "Group",
    "Id": 34,
    "Name": "BP Gold Group"
  }
],
"Type": "User",
"Id": 16
},
{
  "DisplayName": "Ari Test",
  "Name": "aritest",
  "Firstname": "Ari",
  "Lastname": "Test",
  "Groups": [],
  "Type": "User",
  "Id": 17
}
]
```

Get User

```
/api/v1/users/{userId}
```

DESCRIPTION Retrieves the user with the specified user ID.

URL [Blueprint_URI]/api/v1/users/{UserId}
STRUCTURE

- **UserId**: Defines the Id of the user.

SUPPORTED METHODS

- **HEAD**
- **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users/125?
```

Response Header

```
{'content-length': '332', 'pragma': 'no-cache', 'cache-control': 'no-cache',  
'date': 'Sat, 04 May 2013 14:59:22 GMT', 'expires': '-1', 'content-type':  
'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>  
<User xmlns="http://www.blueprintsys.com/blueprint/api/v1"  
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">  
  <DisplayName>API Documentation User</DisplayName>
```

```
<Firstname>API</Firstname>
<Groups>
  <Group>
    <Id>75</Id>
    <Name>API Authors</Name>
  </Group>
</Groups>
<Id>125</Id>
<Lastname>Documentation</Lastname>
<Name>api_docs_user</Name>
</User>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users/125?
```

Response Header

```
{'content-length': '205', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 11:49:50 GMT', 'expires': '-1', 'content-type':
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
{
  "DisplayName": "API Documentation User",
  "Name": "api_docs_user",
  "Firstname": "API",
  "Lastname": "Documentation",
  "Groups": [
    {
      "ProjectId": null,
      "Type": "Group",
      "Id": 75,
      "Name": "API Authors"
    }
  ],
  "Type": "User",
  "Id": 125
}
```


Create User

`/api/v1/users`

DESCRIPTION Creates a user with specific defined properties.

URL [Blueprint_URI]/api/v1/users
STRUCTURE

SUPPORTED METHODS ■ POST

REQUEST BODY The new artifact must be defined in the body of the request. Read more about [how to define a user](#) in the request body.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users
```

Request Body

```
<?xml version="1.0" ?>
<Users xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <User>
    <Email>dquade2084@dev.recall.com</Email>
    <Department>Devops</Department>
    <DisplayName>quaaade</DisplayName>
    <Firstname>Douglas</Firstname>
    <Lastname>Quade</Lastname>
    <Name>dquade</Name>
    <Enabled>>true</Enabled>
    <Password>@Quaaade#1</Password>
```

```
        <ExpiredPassword>>false</ExpiredPassword>
        <GroupIds
xmlns:i="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
            <i:int>8</i:int>
        </GroupIds>
        <Title>Engineer</Title>
    </User>
</Users>
```

Response Body

```
<UserAddResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
    <UserAddResult>
        <Message>User has been created successfully</Message>
        <ResultCode>201</ResultCode>
        <User>
            <DisplayName>quaaade</DisplayName>
            <Enabled>>false</Enabled>
            <Firstname>Douglas</Firstname>
            <Groups />
            <Id>3754</Id>
            <Lastname>Quade</Lastname>
            <Name>dquade</Name>
            <Title>Engineer</Title>
        </User>
    </UserAddResult>
</UserAddResults>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users
```

Request Body

```
[
  {
    "Type": "User",
    "Name": "dquade",
    "DisplayName": "quaaade",
    "Firstname": "Douglas",
    "Lastname": "Quade",
    "GroupIds": [
      "1",
      "8",
      "12"
    ],
    "Title": "Engineer",
    "Department": "Devops",
    "Password": "#fR33m4R5",
    "Enabled": true,
  }
]
```

```
    "Email": "dquade2084@dev.recall.com"  
  }  
]
```

Response Body

```
[  
  {  
    "User": {  
      "Type": "User",  
      "Id": 2084,  
      "Name": "dquade",  
      "DisplayName": "quaaade",  
      "Firstname": "Douglas",  
      "Lastname": "Quade",  
      "GroupIds": [  
        "1",  
        "8",  
        "12"  
      ],  
      "Title": "Engineer",  
      "Department": "Devops",  
      "Password": "#fR33m4R5",  
      "Enabled": true,  
      "Email": "dquade2084@devops.recall.com"  
    },  
    "Message": "User has been created successfully",  
    "ResultCode": 201  
  }  
]
```

Update User

`/api/v1/users`

DESCRIPTION Updates the specified properties of a specific user.

URL [Blueprint_URI]/api/v1/users
STRUCTURE

SUPPORTED METHODS ■ PATCH

REQUEST BODY The artifact must be defined in the body of the request. Read more about [how to define a user](#) in the request body.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users
```

Request Body

(Modifies user title and department created in "create user" example)

```
<?xml version="1.0" ?>
<Users xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <User>
    <Email>dquade2084@dev.recall.com</Email>
    <Department>R&D</Department>
    <Name>dquade</Name>
    <Enabled>>true</Enabled>
    <GroupIds
xmlns:i="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
```

```
        <i:int>1</i:int>
        <i:int>8</i:int>
        <i:int>12</i:int>
    </GroupIds>
    <Title>Senior Engineer</Title>
</User>
</Users>
```

Response Body

```
<UserUpdateResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <UserUpdateResult>
    <Message>User information has been updated successfully</Message>
    <ResultCode>200</ResultCode>
    <User>
      <Email>dquade2084@dev.recall.com</Email>
      <Department>R&D</Department>
      <DisplayName>quaaade</DisplayName>
      <Enabled>>false</Enabled>
      <Firstname>Douglas</Firstname>
      <Groups />
      <Id>3754</Id>
      <Lastname>Quade</Lastname>
      <Name>dquade</Name>
      <Title>Senior Engineer</Title>
    </User>
  </UserUpdateResult>
</UserAddResults>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users
```

Request Body

(Modifies user title and department created in "create user" example)

```
[
  {
    "Type": "User",
    "Name": "dquade",
    "DisplayName": "quaaade",
    "Firstname": "Douglas",
    "Lastname": "Quade",
    "GroupIds": [
      "1",
      "8",
      "12"
    ],
    "Title": "Senior Engineer",
```

```
    "Department": "RandD",  
    "Password": "#fR33m4R5",  
    "Enabled": true,  
    "Email": "dquade2084@dev.recall.com"  
  }  
]
```

Response Body

```
[  
  {  
    "User": {  
      "Type": "User",  
      "Id": 1040,  
      "Name": "dquade_jtest",  
      "DisplayName": "quaaade_jtest",  
      "Firstname": "Douglas",  
      "Lastname": "Quade",  
      "Groups": [  
        {  
          "Type": "Group",  
          "Id": 1,  
          "Name": "authorized"  
        },  
        {  
          "Type": "Group",  
          "Id": 8,  
          "Name": "Gen Mgrs"  
        },  
        {  
          "Type": "Group",  
          "Id": 12,  
          "Name": "sysgen group 1"  
        }  
      ],  
      "Title": "Senior Engineer",  
      "Department": "RandD",  
      "Enabled": true,  
      "Email": "dquade2084@dev.recall.com"  
    },  
    "Message": "User information has been updated successfully",  
    "ResultCode": 200  
  }  
]
```

Delete User

`/api/v1/users`

DESCRIPTION Deletes all specified users.

URL `[Blueprint_URI]/api/v1/users`
STRUCTURE

SUPPORTED METHODS ■ DELETE

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users
```

Request Body

```
<?xml version="1.0" ?>
<Users xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <User>
    <DisplayName>quaaade</DisplayName>
    <Firstname>Douglas</Firstname>
    <Lastname>Quade</Lastname>
    <Name>dquade</Name>
    <Password>@Quaaade#1</Password>
  </User>
</Users>
```

Response Body

```
<UserAddResults xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <UserAddResult>
    <Message>User has been successfully deleted.</Message>
    <ResultCode>200</ResultCode>
    <User>
      <Email>dquade2084@dev.recall.com</Email>
      <Department>Devops</Department>
      <DisplayName>quaaade</DisplayName>
      <Enabled>>false</Enabled>
      <Firstname>Douglas</Firstname>
      <Groups />
      <Id>3754</Id>
      <Lastname>Quade</Lastname>
      <Name>dquade</Name>
      <Title>Engineer</Title>
    </User>
  </UserAddResult>
</UserAddResults>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/users
```

Request Body

```
[
  "Fanvid",
  "Noviseth"
]
```

Response Body

```
[
  {
    "User": {
      "Type": "User",
      "Id": 22,
      "Name": "Fanvid",
      "DisplayName": "David",
      "Firstname": "David",
      "Lastname": "Gao",
      "Title": "Tester",
      "Department": "RD",
      "Enabled": true,
      "Email": "WXZALZ3RSK@FLQAPZON90.com"
    },
    "Message": "User has been successfully deleted.",
    "ResultCode": 200
  },
  {
```



```
    "User": {  
      "Type": "User",  
      "Id": 31,  
      "Name": "noviseth",  
      "DisplayName": "Novina Sethuraman",  
      "Firstname": "Novina",  
      "Lastname": "Sethuraman",  
      "Title": "Tester",  
      "Department": "RD",  
      "Enabled": true,  
      "Email": "4SLH4FQOSA@M7D01B7OB0.com"  
    },  
    "Message": "User has been successfully deleted.",  
    "ResultCode": 200  
  }  
]
```

List Artifact Types

```
/api/v1/projects/{projectId}/metadata/artifactTypes?propertyTypes={boolean}&richtextasplain={boolean}&inlinecss={boolean}&offset={offset}&limit={limit}
```

DESCRIPTION Retrieves all artifact types available in the specified project.

Note: The `IsUsedInThisProject`, `StandardPropertyTypeId`, and `StandardArtifactTypeId` elements in the response body are only returned when using Blueprint 7.3 and later.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/metadata/ArtifactTypes

SUPPORTED METHODS

- HEAD
- GET

REQUEST URI PARAMETERS

- `PropertyTypes` (optional): Defines whether or not to include property types. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- `RichTextAsPlain` (optional): Defines whether or not to retrieve all rich-text properties as Plain Text instead of HTML. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- `InlineCSS` (optional): Defines whether or not to retrieve all rich-text properties with locally defined or inline styles. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value. When this parameter is set to **false**, rich-text properties return internal styles that are defined within the `<head>` section of the HTML.

Note: This parameter is supported in Blueprint 6.1 and later.

- `Offset` (optional): Defines the paging offset (that is, index) at which the results start. Offset the default is **0**.
- `Limit` (optional): Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is **100**. The minimum value is **1** and the maximum value is **500**.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/metadata/artifactTypes?Limit=5&Offset=0
```

Response Header

```
{'content-length': '1394', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'artifact_types 0-4/22', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 14:59:12 GMT', 'content-type': 'application/xml; charset=utf-8'}
```

Response Body

```
<?xml version="1.0" ?>
  <ArtifactTypes xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <ArtifactType>
    <BaseArtifactType>PrimitiveFolder</BaseArtifactType>
    <Description>Description</Description>
    <Id>551</Id>
    <Name>Folder</Name>
    <Prefix>PF</Prefix>
  </ArtifactType>
  <ArtifactType>
    <BaseArtifactType>Glossary</BaseArtifactType>
    <Description>List of terms and definitions.</Description>
    <Id>552</Id>
    <Name>Glossary</Name>
    <Prefix>GL</Prefix>
  </ArtifactType>
  <ArtifactType>
    <BaseArtifactType>TextualRequirement</BaseArtifactType>
    <Description>A text based requirement that does not fit into any of the
other textual requirement types.</Description>
    <Id>553</Id>
    <Name>Textual Requirement</Name>
    <Prefix>RQ</Prefix>
  </ArtifactType>
  <ArtifactType>
    <BaseArtifactType>BusinessProcess</BaseArtifactType>
    <Description>A swimlane oriented process diagram depicting how business
```

```
tasks relate one to another to accomplish an overall goal of the
business.</Description>
  <Id>554</Id>
  <Name>Business Process Diagram</Name>
  <Prefix>BP-DG</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Actor</BaseArtifactType>
  <Description>An actor is either a human or another system that
interacts with the system under development for the purpose of accomplishing
certain goals (use cases) with the system.</Description>
  <Id>555</Id>
  <Name>Actor</Name>
  <Prefix>ACTOR</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>UseCase</BaseArtifactType>
  <Description>Describes the interaction between an actor and the system
under development to accomplish a given goal.</Description>
  <Id>556</Id>
  <Name>Use Case</Name>
  <Prefix>UC</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>UIMockup</BaseArtifactType>
  <Description>A user interface mockup of a screen for the system under
developent.</Description>
  <Id>557</Id>
  <Name>Screen</Name>
  <Prefix>SCRN</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>GenericDiagram</BaseArtifactType>
  <Description>Any diagram format that is not a Business Process or
Domain diagram.</Description>
  <Id>558</Id>
  <Name>Generic Diagram</Name>
  <Prefix>GEN-DG</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Document</BaseArtifactType>
  <Description>Description</Description>
  <Id>559</Id>
  <Name>Document</Name>
  <Prefix>DOC</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Storyboard</BaseArtifactType>
  <Description>A connected set of screens used to demonstrate screen-to-
screen based navigation.</Description>
  <Id>560</Id>
  <Name>Storyboard</Name>
  <Prefix>SB</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>DomainDiagram</BaseArtifactType>
  <Description>A diagram depicting how entities or classes relate to one
another.</Description>
  <Id>561</Id>
  <Name>Domain Diagram</Name>
```

```
<Prefix>DOM-DG</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Describes a business need and is considered to answer the
question "why" does the business need this.</Description>
  <Id>565</Id>
  <Name>Business Requirement</Name>
  <Prefix>BUS-RQ</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Describes a business constraint that the system under
development must adhere to.</Description>
  <Id>566</Id>
  <Name>Business Rule</Name>
  <Prefix>BUS-RL</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Defines the characteristics (type, length, etc.) of data
elements.</Description>
  <Id>567</Id>
  <Name>Data Definitions</Name>
  <Prefix>DATA-DEF</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Describes the tasks that the system must perform in order
to satisfy the tasks of the users. Typically answers the question "how" will
the system meet the needs of the users.</Description>
  <Id>568</Id>
  <Name>Functional Requirement</Name>
  <Prefix>FUNC-RQ</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Describes the non-functional characteristics of the
system. Often referred to as the "ilities," e.g. scalability, reliability,
usability, etc.</Description>
  <Id>569</Id>
  <Name>Quality of Service Requirement</Name>
  <Prefix>QS-RQ</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Defines the specific fields for a given
screen.</Description>
  <Id>570</Id>
  <Name>Screen Definitions</Name>
  <Prefix>SCRN-DEF</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Describes the tasks that a user needs to perform with the
system in order to satisfy the needs of the business. Typically answers the
question "what" must the user do with the system.</Description>
  <Id>571</Id>
  <Name>User Requirement</Name>
  <Prefix>USER-RQ</Prefix>
```

```

</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Used to document error messages and
warnings.</Description>
  <Id>572</Id>
  <Name>Error Messages</Name>
  <Prefix>ERR-MSG</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>UseCaseDiagram</BaseArtifactType>
  <Description>Description</Description>
  <Id>573</Id>
  <Name>Use Case Diagram</Name>
  <Prefix>UCD</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Edit Description</Description>
  <Id>574</Id>
  <Name>Information</Name>
  <Prefix>INFO</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Process</BaseArtifactType>
  <Description>Description</Description>
  <Id>3532</Id>
  <Name>Process</Name>
  <Prefix>PRO</Prefix>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Used for Storyteller</Description>
  <Id>4019</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>ST-User Story(6)</Name>
  <Prefix>STUS(7)</Prefix>
  <StandardArtifactTypeId>4016</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Epic is a large User Story</Description>
  <Id>4041</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Epic(Agile Pack)</Name>
  <Prefix>EP(Agile Pack)</Prefix>
  <StandardArtifactTypeId>4038</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>A process that adds value to a customer and it can be
planned.</Description>
  <Id>4063</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Feature(Agile Pack)</Name>
  <Prefix>FT(Agile Pack)</Prefix>
  <StandardArtifactTypeId>4060</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>

```

```
<Description>A real-world example of how one or more people or
organizations interact with a system.</Description>
<Id>4085</Id>
<IsUsedInThisProject>true</IsUsedInThisProject>
<Name>Scenario (Agile Pack)</Name>
<Prefix>SC (Agile Pack)</Prefix>
<StandardArtifactTypeId>4082</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>A grouping of Stories.</Description>
  <Id>4107</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Theme (Agile Pack)</Name>
  <Prefix>TH (Agile Pack)</Prefix>
  <StandardArtifactTypeId>4104</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Used in agile development to capture the description of a
software feature from an end-user perspective.</Description>
  <Id>4129</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>ST-User Story (Agile Pack)</Name>
  <Prefix>ST-US (Agile Pack)</Prefix>
  <StandardArtifactTypeId>4126</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>UseCase</BaseArtifactType>
  <Description>
</Description>
  <Id>4538</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 1</Name>
  <Prefix>Prefix 1</Prefix>
  <StandardArtifactTypeId>4535</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>UseCase</BaseArtifactType>
  <Description>
</Description>
  <Id>4590</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 3</Name>
  <Prefix>Prefix 3</Prefix>
  <StandardArtifactTypeId>4587</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Document</BaseArtifactType>
  <Description>
</Description>
  <Id>4616</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 4</Name>
  <Prefix>Prefix 4</Prefix>
  <StandardArtifactTypeId>4613</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Actor</BaseArtifactType>
  <Description>
```

```
</Description>
<Id>4642</Id>
<IsUsedInThisProject>>true</IsUsedInThisProject>
<Name>Custom V1 Artifact</Name>
<Prefix>CVA</Prefix>
<StandardArtifactTypeId>4639</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Actor</BaseArtifactType>
  <Description>
  </Description>
  <Id>5176</Id>
  <IsUsedInThisProject>>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 5</Name>
  <Prefix>Prefix 5</Prefix>
  <StandardArtifactTypeId>5173</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>PrimitiveFolder</BaseArtifactType>
  <Description>
  </Description>
  <Id>5207</Id>
  <IsUsedInThisProject>>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 6</Name>
  <Prefix>Prefix 6</Prefix>
  <StandardArtifactTypeId>5204</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Glossary</BaseArtifactType>
  <Description>
  </Description>
  <Id>5238</Id>
  <IsUsedInThisProject>>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 7</Name>
  <Prefix>Prefix 7</Prefix>
  <StandardArtifactTypeId>5235</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Actor</BaseArtifactType>
  <Description>
  </Description>
  <Id>5269</Id>
  <IsUsedInThisProject>>true</IsUsedInThisProject>
  <Name>Standard Artifact Type 8</Name>
  <Prefix>Prefix 8</Prefix>
  <StandardArtifactTypeId>5266</StandardArtifactTypeId>
</ArtifactType>
<ArtifactType>
  <BaseArtifactType>Process</BaseArtifactType>
  <Description>
  </Description>
  <Id>5380</Id>
  <IsUsedInThisProject>>true</IsUsedInThisProject>
  <Name>Process (STD)</Name>
  <Prefix>PRO_STD</Prefix>
  <StandardArtifactTypeId>5377</StandardArtifactTypeId>
</ArtifactType>
</ArtifactTypes>
```


JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/metadata/artifactTypes?Limit=5&Offset=0
```

Response Header

```
{'content-length': '944', 'expires': '-1', 'server': 'Blueprint', 'content-range': 'artifact_types 0-4/22', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 11:42:57 GMT', 'content-type': 'application/json; charset=utf-8'}
```

Response Body

```
[
  {
    "Id": 551,
    "Name": "Folder",
    "Description": "Description",
    "Prefix": "PF",
    "BaseArtifactType": "PrimitiveFolder"
  },
  {
    "Id": 552,
    "Name": "Glossary",
    "Description": "List of terms and definitions.",
    "Prefix": "GL",
    "BaseArtifactType": "Glossary"
  },
  {
    "Id": 553,
    "Name": "Textual Requirement",
    "Description": "A text based requirement that does not fit into any of the other textual requirement types.",
    "Prefix": "RQ",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 554,
    "Name": "Business Process Diagram",
    "Description": "A swimlane oriented process diagram depicting how business tasks relate one to another to accomplish an overall goal of the business.",
    "Prefix": "BP-DG",
    "BaseArtifactType": "BusinessProcess"
  },
  {
    "Id": 555,
    "Name": "Actor",
    "Description": "An actor is either a human or another system that interacts with the system under development for the purpose of accomplishing certain goals (use cases) with the system.",
    "Prefix": "ACTOR",
  }
]
```

```
    "BaseArtifactType": "Actor"
  },
  {
    "Id": 556,
    "Name": "Use Case",
    "Description": "Describes the interaction between an actor and the system
under development to accomplish a given goal.",
    "Prefix": "UC",
    "BaseArtifactType": "UseCase"
  },
  {
    "Id": 557,
    "Name": "Screen",
    "Description": "A user interface mockup of a screen for the system under
developent.",
    "Prefix": "SCRN",
    "BaseArtifactType": "UIMockup"
  },
  {
    "Id": 558,
    "Name": "Generic Diagram",
    "Description": "Any diagram format that is not a Business Process or Domain
diagram.",
    "Prefix": "GEN-DG",
    "BaseArtifactType": "GenericDiagram"
  },
  {
    "Id": 559,
    "Name": "Document",
    "Description": "Description",
    "Prefix": "DOC",
    "BaseArtifactType": "Document"
  },
  {
    "Id": 560,
    "Name": "Storyboard",
    "Description": "A connected set of screens used to demonstrate screen-to-
screen based navigation.",
    "Prefix": "SB",
    "BaseArtifactType": "Storyboard"
  },
  {
    "Id": 561,
    "Name": "Domain Diagram",
    "Description": "A diagram depicting how entities or classes relate to one
another.",
    "Prefix": "DOM-DG",
    "BaseArtifactType": "DomainDiagram"
  },
  {
    "Id": 565,
    "Name": "Business Requirement",
    "Description": "Describes a business need and is considered to answer the
question \"why\" does the business need this.",
    "Prefix": "BUS-RQ",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 566,
    "Name": "Business Rule",
```

```
    "Description": "Describes a business constraint that the system under
development must adhere to.",
    "Prefix": "BUS-RL",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 567,
    "Name": "Data Definitions",
    "Description": "Defines the characteristics (type, length, etc.) of data
elements.",
    "Prefix": "DATA-DEF",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 568,
    "Name": "Functional Requirement",
    "Description": "Describes the tasks that the system must perform in order
to satisfy the tasks of the users. Typically answers the question \"how\" will
the system meet the needs of the users.",
    "Prefix": "FUNC-RQ",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 569,
    "Name": "Quality of Service Requirement",
    "Description": "Describes the non-functional characteristics of the system.
Often referred to as the \"ilities,\" e.g. scalability, reliability, usability,
etc.",
    "Prefix": "QS-RQ",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 570,
    "Name": "Screen Definitions",
    "Description": "Defines the specific fields for a given screen.",
    "Prefix": "SCRN-DEF",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 571,
    "Name": "User Requirement",
    "Description": "Describes the tasks that a user needs to perform with the
system in order to satisfy the needs of the business. Typically answers the
question \"what\" must the user do with the system.",
    "Prefix": "USER-RQ",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 572,
    "Name": "Error Messages",
    "Description": "Used to document error messages and warnings.",
    "Prefix": "ERR-MSG",
    "BaseArtifactType": "TextualRequirement"
  },
  {
    "Id": 573,
    "Name": "Use Case Diagram",
    "Description": "Description",
    "Prefix": "UCD",
    "BaseArtifactType": "UseCaseDiagram"
  }
}
```

```
    },
    {
      "Id": 574,
      "Name": "Information",
      "Description": "Edit Description",
      "Prefix": "INFO",
      "BaseArtifactType": "TextualRequirement"
    },
    {
      "Id": 3532,
      "Name": "Process",
      "Description": "Description",
      "Prefix": "PRO",
      "BaseArtifactType": "Process"
    },
    {
      "Id": 4019,
      "Name": "ST-User Story(6)",
      "Description": "Used for Storyteller",
      "Prefix": "STUS(7)",
      "BaseArtifactType": "TextualRequirement",
      "StandardArtifactTypeId": 4016,
      "IsUsedInThisProject": true
    },
    {
      "Id": 4041,
      "Name": "Epic(Agile Pack)",
      "Description": "Epic is a large User Story",
      "Prefix": "EP(Agile Pack)",
      "BaseArtifactType": "TextualRequirement",
      "StandardArtifactTypeId": 4038,
      "IsUsedInThisProject": true
    },
    {
      "Id": 4063,
      "Name": "Feature(Agile Pack)",
      "Description": "A process that adds value to a customer and it can be
planned.",
      "Prefix": "FT(Agile Pack)",
      "BaseArtifactType": "TextualRequirement",
      "StandardArtifactTypeId": 4060,
      "IsUsedInThisProject": true
    },
    {
      "Id": 4085,
      "Name": "Scenario(Agile Pack)",
      "Description": "A real-world example of how one or more people or
organizations interact with a system.",
      "Prefix": "SC(Agile Pack)",
      "BaseArtifactType": "TextualRequirement",
      "StandardArtifactTypeId": 4082,
      "IsUsedInThisProject": true
    },
    {
      "Id": 4107,
      "Name": "Theme(Agile Pack)",
      "Description": "A grouping of Stories.",
      "Prefix": "TH(Agile Pack)",
      "BaseArtifactType": "TextualRequirement",
      "StandardArtifactTypeId": 4104,
```

```
    "IsUsedInThisProject": true
  },
  {
    "Id": 4129,
    "Name": "ST-User Story(Agile Pack)",
    "Description": "Used in agile development to capture the description of a
software feature from and end-user perspective.",
    "Prefix": "ST-US(Agile Pack)",
    "BaseArtifactType": "TextualRequirement",
    "StandardArtifactTypeId": 4126,
    "IsUsedInThisProject": true
  },
  {
    "Id": 4538,
    "Name": "Standard Artifact Type 1",
    "Description": "",
    "Prefix": "Prefix 1",
    "BaseArtifactType": "UseCase",
    "StandardArtifactTypeId": 4535,
    "IsUsedInThisProject": true
  },
  {
    "Id": 4590,
    "Name": "Standard Artifact Type 3",
    "Description": "",
    "Prefix": "Prefix 3",
    "BaseArtifactType": "UseCase",
    "StandardArtifactTypeId": 4587,
    "IsUsedInThisProject": true
  },
  {
    "Id": 4616,
    "Name": "Standard Artifact Type 4",
    "Description": "",
    "Prefix": "Prefix 4",
    "BaseArtifactType": "Document",
    "StandardArtifactTypeId": 4613,
    "IsUsedInThisProject": true
  },
  {
    "Id": 4642,
    "Name": "Custom V1 Artifact",
    "Description": "",
    "Prefix": "CVA",
    "BaseArtifactType": "Actor",
    "StandardArtifactTypeId": 4639,
    "IsUsedInThisProject": true
  },
  {
    "Id": 5176,
    "Name": "Standard Artifact Type 5",
    "Description": "",
    "Prefix": "Prefix 5",
    "BaseArtifactType": "Actor",
    "StandardArtifactTypeId": 5173,
    "IsUsedInThisProject": true
  },
  {
    "Id": 5207,
    "Name": "Standard Artifact Type 6",
```

```
"Description": "",
"Prefix": "Prefix 6",
"BaseArtifactType": "PrimitiveFolder",
"StandardArtifactTypeId": 5204,
"IsUsedInThisProject": true
},
{
  "Id": 5238,
  "Name": "Standard Artifact Type 7",
  "Description": "",
  "Prefix": "Prefix 7",
  "BaseArtifactType": "Glossary",
  "StandardArtifactTypeId": 5235,
  "IsUsedInThisProject": true
},
{
  "Id": 5269,
  "Name": "Standard Artifact Type 8",
  "Description": "",
  "Prefix": "Prefix 8",
  "BaseArtifactType": "Actor",
  "StandardArtifactTypeId": 5266,
  "IsUsedInThisProject": true
},
{
  "Id": 5380,
  "Name": "Process (STD)",
  "Description": "",
  "Prefix": "PRO_STD",
  "BaseArtifactType": "Process",
  "StandardArtifactTypeId": 5377,
  "IsUsedInThisProject": true
}
]
```

Get Artifact Type

```
/api/v1/projects/{projectId}/metadata/artifactTypes/  
{artifactTypeId}?richtextasplain={richtextasplain}&inlinecss={inlinecss}
```

DESCRIPTION Retrieves all artifact type objects from a project.

Note: The `IsUsedInThisProject`, `StandardPropertyTypeId`, and `StandardArtifactTypeId` elements in the response body are only returned when using Blueprint 7.3 and later.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/
{ProjectId}/metadata/artifacttypes/{ArtifactTypeId}

- `ProjectId`: The ID of the project that contains the artifacts you want to get.
- `ArtifactTypeId`: The ID of the artifact types you want to get.

SUPPORTED METHODS

- HEAD
- GET

REQUEST URI
PARAMETERS

- `RichTextAsPlain` (optional): Defines whether or not to retrieve all rich-text properties as Plain Text instead of HTML. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value.
- `InlineCSS` (optional): Defines whether or not to retrieve all rich-text properties with locally defined or inline styles. This parameter can be set to **true** or **false**. The default is **false** if not specified. The default is **true** if the parameter is included in the URI with no given value. When this parameter is set to **false**, rich-text properties return internal styles that are defined within the `<head>` section of the HTML.

Note: This parameter is supported in Blueprint 6.1 and later.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/metadata/artifactTypes/11559?
```

Response Header

```
{'content-length': '4336', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Sat, 04 May 2013 14:59:14 GMT', 'expires': '-1', 'content-type': 'application/xml; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
<?xml version="1.0" ?>
<ArtifactType xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <BaseArtifactType>TextualRequirement</BaseArtifactType>
  <Description>Epic is a large User Story</Description>
  <Id>11559</Id>
  <IsUsedInThisProject>true</IsUsedInThisProject>
  <Name>Epic (Agile Pack)</Name>
  <Prefix>EP (Agile Pack)</Prefix>
  <PropertyTypes>
    <PropertyType i:type="TextPropertyType">
      <BasePropertyType>Text</BasePropertyType>
      <HasDefaultValue>>false</HasDefaultValue>
      <Id>434</Id>
      <IsReadOnly>true</IsReadOnly>
      <IsRequired>>false</IsRequired>
      <Name>ID</Name>
      <IsMultiLine>>false</IsMultiLine>
      <IsRichText>>false</IsRichText>
    </PropertyType>
    <PropertyType i:type="TextPropertyType">
      <BasePropertyType>Text</BasePropertyType>
      <HasDefaultValue>>false</HasDefaultValue>
      <Id>435</Id>
      <IsReadOnly>>false</IsReadOnly>
      <IsRequired>true</IsRequired>
      <Name>Name</Name>
      <IsMultiLine>>false</IsMultiLine>
      <IsRichText>>false</IsRichText>
    </PropertyType>
    <PropertyType i:type="TextPropertyType">
      <BasePropertyType>Text</BasePropertyType>
      <HasDefaultValue>>true</HasDefaultValue>
      <Id>436</Id>
      <IsReadOnly>>false</IsReadOnly>
```



```
<IsRequired>>false</IsRequired>
<Name>Description</Name>
<DefaultValue><html><head></head><body style="padding: 1px 0px 0px; font-
family: 'Portable User Interface'; font-size: 10.67px"><div style="padding:
0px"><p style="margin: 0px">&#x200b;</p></div></body></html></DefaultValue>
<IsMultiLine>>true</IsMultiLine>
<IsRichText>>true</IsRichText>
</PropertyType>
<PropertyType i:type="UserPropertyType">
  <BasePropertyType>User</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>474</Id>
  <IsReadOnly>>true</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Created By</Name>
</PropertyType>
<PropertyType i:type="DatePropertyType">
  <BasePropertyType>Date</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>475</Id>
  <IsReadOnly>>true</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Created On</Name>
  <IsValidated>>false</IsValidated>
</PropertyType>
<PropertyType i:type="UserPropertyType">
  <BasePropertyType>User</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>476</Id>
  <IsReadOnly>>true</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Last Edited By</Name>
</PropertyType>
<PropertyType i:type="DatePropertyType">
  <BasePropertyType>Date</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>477</Id>
  <IsReadOnly>>true</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Last Edited On</Name>
  <IsValidated>>false</IsValidated>
</PropertyType>
<PropertyType i:type="ChoicePropertyType">
  <BasePropertyType>Choice</BasePropertyType>
  <HasDefaultValue>>true</HasDefaultValue>
  <Id>3881</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Tshirt (Agile Pack)</Name>
  <StandardPropertyTypeId>3878</StandardPropertyTypeId>
  <AllowMultiple>>false</AllowMultiple>
  <DefaultValueIndex>0</DefaultValueIndex>
  <IsCustomValueAllowed>>false</IsCustomValueAllowed>
  <ValidValues>
    <ValidValue>XS</ValidValue>
    <ValidValue>S</ValidValue>
    <ValidValue>M</ValidValue>
    <ValidValue>L</ValidValue>
    <ValidValue>XL</ValidValue>
  </ValidValues>
```

```
</PropertyType>
<PropertyType i:type="TextPropertyType">
  <BasePropertyType>Text</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>4875</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>22</Name>
  <StandardPropertyTypeId>4872</StandardPropertyTypeId>
  <IsMultiLine>>true</IsMultiLine>
  <IsRichText>>true</IsRichText>
</PropertyType>
<PropertyType i:type="TextPropertyType">
  <BasePropertyType>Text</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>4905</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Standard Property 6</Name>
  <StandardPropertyTypeId>4902</StandardPropertyTypeId>
  <IsMultiLine>>true</IsMultiLine>
  <IsRichText>>true</IsRichText>
</PropertyType>
<PropertyType i:type="TextPropertyType">
  <BasePropertyType>Text</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>5036</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Standard Property 7</Name>
  <StandardPropertyTypeId>5033</StandardPropertyTypeId>
  <IsMultiLine>>true</IsMultiLine>
  <IsRichText>>true</IsRichText>
</PropertyType>
<PropertyType i:type="TextPropertyType">
  <BasePropertyType>Text</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>5099</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Standard Property 9</Name>
  <StandardPropertyTypeId>5096</StandardPropertyTypeId>
  <IsMultiLine>>true</IsMultiLine>
  <IsRichText>>true</IsRichText>
</PropertyType>
<PropertyType i:type="TextPropertyType">
  <BasePropertyType>Text</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>5130</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>>false</IsRequired>
  <Name>Standard Property 10</Name>
  <StandardPropertyTypeId>5127</StandardPropertyTypeId>
  <IsMultiLine>>true</IsMultiLine>
  <IsRichText>>true</IsRichText>
</PropertyType>
<PropertyType i:type="TextPropertyType">
  <BasePropertyType>Text</BasePropertyType>
  <HasDefaultValue>>false</HasDefaultValue>
  <Id>5298</Id>
```

```
<IsReadOnly>>false</IsReadOnly>
<IsRequired>true</IsRequired>
<Name>Standard Property 12</Name>
<StandardPropertyTypeId>5295</StandardPropertyTypeId>
<IsMultiLine>true</IsMultiLine>
<IsRichText>true</IsRichText>
</PropertyType>
<PropertyType i:type="ChoicePropertyType">
  <BasePropertyType>Choice</BasePropertyType>
  <HasDefaultValue>true</HasDefaultValue>
  <Id>6676</Id>
  <IsReadOnly>>false</IsReadOnly>
  <IsRequired>false</IsRequired>
  <Name>Standard Property 13</Name>
  <StandardPropertyTypeId>6673</StandardPropertyTypeId>
  <AllowMultiple>true</AllowMultiple>
  <DefaultValueIndex>0</DefaultValueIndex>
  <IsCustomValueAllowed>>false</IsCustomValueAllowed>
  <ValidValues>
    <ValidValue>a</ValidValue>
    <ValidValue>b</ValidValue>
  </ValidValues>
</PropertyType>
</PropertyTypes>
<StandardArtifactTypeId>4038</StandardArtifactTypeId>
</ArtifactType>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/metadata/artifactTypes/11559?
```

Response Header

```
{'content-length': '2240', 'pragma': 'no-cache', 'cache-control': 'no-cache',
'date': 'Sat, 04 May 2013 11:42:58 GMT', 'expires': '-1', 'content-type':
'application/json; charset=utf-8', 'server': 'Blueprint'}
```

Response Body

```
{
  "Id": 11559,
  "Name": "Epic (Agile Pack)",
  "Description": "Epic is a large User Story",
  "Prefix": "EP (Agile Pack)",
  "BaseArtifactType": "TextualRequirement",
  "StandardArtifactTypeId": 4038,
  "IsUsedInThisProject": true,
  "PropertyTypes":
  [
    {
```

```
    "BasePropertyType": "Text",
    "IsRichText": false,
    "IsMultiLine": false,
    "Id": 434,
    "Name": "ID",
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": true
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": false,
    "IsMultiLine": false,
    "Id": 435,
    "Name": "Name",
    "IsRequired": true,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": true,
    "IsMultiLine": true,
    "DefaultValue": "?",
    "Id": 436,
    "Name": "Description",
    "IsRequired": false,
    "HasDefaultValue": true,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "User",
    "Id": 474,
    "Name": "Created By",
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": true
  },
  {
    "BasePropertyType": "Date",
    "IsValidated": false,
    "Id": 475,
    "Name": "Created On",
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": true
  },
  {
    "BasePropertyType": "User",
    "Id": 476,
    "Name": "Last Edited By",
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": true
  },
  {
    "BasePropertyType": "Date",
    "IsValidated": false,
    "Id": 477,
    "Name": "Last Edited On",
```

```
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": true
  },
  {
    "BasePropertyType": "Choice",
    "IsCustomValueAllowed": false,
    "AllowMultiple": false,
    "ValidValues":
    [
      "",
      "XS",
      "S",
      "M",
      "L",
      "XL"
    ],
    "DefaultValueIndex": 0,
    "Id": 3881,
    "Name": "Tshirt (Agile Pack)",
    "StandardPropertyTypeId": 3878,
    "IsRequired": false,
    "HasDefaultValue": true,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": true,
    "IsMultiLine": true,
    "Id": 4875,
    "Name": "22",
    "StandardPropertyTypeId": 4872,
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": true,
    "IsMultiLine": true,
    "Id": 4905,
    "Name": "Standard Property 6",
    "StandardPropertyTypeId": 4902,
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": true,
    "IsMultiLine": true,
    "Id": 5036,
    "Name": "Standard Property 7",
    "StandardPropertyTypeId": 5033,
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
```

```
    "IsRichText": true,
    "IsMultiLine": true,
    "Id": 5099,
    "Name": "Standard Property 9",
    "StandardPropertyTypeId": 5096,
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": true,
    "IsMultiLine": true,
    "Id": 5130,
    "Name": "Standard Property 10",
    "StandardPropertyTypeId": 5127,
    "IsRequired": false,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Text",
    "IsRichText": true,
    "IsMultiLine": true,
    "Id": 5298,
    "Name": "Standard Property 12",
    "StandardPropertyTypeId": 5295,
    "IsRequired": true,
    "HasDefaultValue": false,
    "IsReadOnly": false
  },
  {
    "BasePropertyType": "Choice",
    "IsCustomValueAllowed": false,
    "AllowMultiple": true,
    "ValidValues":
    [
      "a",
      "b"
    ],
    "DefaultValueIndex": 0,
    "Id": 6676,
    "Name": "Standard Property 13",
    "StandardPropertyTypeId": 6673,
    "IsRequired": false,
    "HasDefaultValue": true,
    "IsReadOnly": false
  }
]
}
```

Get Project Discussion Status Values

Note: This request is supported in Blueprint 5.4 and later.

`/api/v1/projects/{projectId}/metadata/discussionStatusValues`

DESCRIPTION Retrieves a list of discussion status values that are valid for the specified project.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/metadata/discussionStatusValues

- **ProjectId:** The ID of the project that contains the discussion status values you want to retrieve.

SUPPORTED METHODS

- **HEAD**
- **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/metadata/discussionStatusValues
```

Response Body

```
<?xml version="1.0" ?>
<DiscussionStatusValues xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <DiscussionStatusValue>
    <Id>100</Id>
    <IsClosed>>false</IsClosed>
    <IsPredefined>>true</IsPredefined>
    <OrderIndex>0</OrderIndex>
  </DiscussionStatusValue>
</DiscussionStatusValues>
```

```
<Status>Open</Status>
</DiscussionStatusValue>
<DiscussionStatusValue>
  <Id>101</Id>
  <IsClosed>true</IsClosed>
  <IsPredefined>true</IsPredefined>
  <OrderIndex>1</OrderIndex>
  <Status>Closed</Status>
</DiscussionStatusValue>
<DiscussionStatusValue>
  <Id>102</Id>
  <IsClosed>false</IsClosed>
  <IsPredefined>false</IsPredefined>
  <OrderIndex>2</OrderIndex>
  <Status>My Open Status</Status>
</DiscussionStatusValue>
</DiscussionStatusValues>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/metadata/discussionStatusValues
```

Response Body

```
[
  {
    "Id": 100,
    "OrderIndex": 0,
    "IsPredefined": true,
    "Status": "Open",
    "IsClosed": false
  },
  {
    "Id": 101,
    "OrderIndex": 1,
    "IsPredefined": true,
    "Status": "Closed",
    "IsClosed": true
  },
  {
    "Id": 102,
    "OrderIndex": 2,
    "IsPredefined": false,
    "Status": "My Open Status",
    "IsClosed": false
  }
]
```


Update Standard Choice-Property Type

Note: This request is supported in Blueprint 7.3 and later.

`/api/v1/instance/metadata/standardChoicePropertyTypes`

DESCRIPTION Updates the standard (instance-level) choice-type property, including property name, valid values, and default value.

URL STRUCTURE [Blueprint_URI]/api/v1/instance/metadata/standardChoicePropertyTypes

SUPPORTED METHODS ■ [PATCH](#)

Note: You can [override the PATCH method](#) and use the [POST method](#) instead.

REQUEST BODY The list of artifacts that you want to publish must be included in the body of the request. Read more about how to specify a [list of artifacts](#) in the request body.

Blueprint result codes

Your response may contain `resultCode` and `message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - The artifact updated successfully.
- **400 (Bad Request)**
- **403: Forbidden**
 - The applicable permissions are needed. For example, this error can occur when:
 - An Author or Collaborate license is needed for access.
 - Comment permissions are needed on the artifact.
 - Edit permissions are needed on the change summary job.
- **404 (Not Found)**
 - Artifact not found in the project.
 - Artifact type not found.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/instance/metadata/standardChoicePropertyTypes
```

Request Body

```
<?xml version="1.0" ?>
<ChoicePropertyType xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Id>3878</Id>
  <Name>Tshirt</Name>
  <DefaultValueIndex>1</DefaultValueIndex>
  <ValidValues>
    <ValidValue>XS</ValidValue>
    <ValidValue>S</ValidValue>
    <ValidValue>M</ValidValue>
    <ValidValue>L</ValidValue>
    <ValidValue>XL</ValidValue>
  </ValidValues>
</ChoicePropertyType>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/instance/metadata/standardChoicePropertyTypes
```

Request Body

```
[
  {
    "Id": 3878,
    "Name": "Tshirt",
    "DefaultValueIndex": "1",
    "ValidValues": [
      "XS",
```

```
"S",  
"M",  
"L",  
"XLWaiting"  
]  
}  
]
```

Update Custom Choice-Property Type

Note: This request is supported in Blueprint 7.3 and later.

```
/api/v1/projects/{projectId}/metadata/choicePropertyTypes
```

DESCRIPTION Updates the custom (project-level) choice-type property, including custom property name, valid values, and default value.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/metadata/choicePropertyTypes

- **ProjectId:** The ID of the project that contains the custom property you want to update or create.

SUPPORTED METHODS ■ **PATCH**

Note: You can [override the PATCH method](#) and use the [POST method](#) instead.

REQUEST BODY The list of artifacts that you want to publish must be included in the body of the request. Read more about how to specify a [list of artifacts](#) in the request body.

Blueprint result codes

Your response may contain `ResultCode` and `Message` parameters if your request involves changes to Blueprint data. These parameters provide detailed information about the result of your request.

- **200 (OK)**
 - The artifact updated successfully.
- **400 (Bad Request)**
- **403: Forbidden**
 - The applicable permissions are needed. For example, this error can occur when:
 - An Author or Collaborate license is needed for access.
 - Comment permissions are needed on the artifact.
 - Edit permissions are needed on the change summary job.
- **404 (Not Found)**

- Artifact not found in the project.
- Artifact type not found.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/45921/metadata/choicePropertyTypes
```

Request Body

```
<?xml version="1.0" ?>
<ChoicePropertyType xmlns="http://www.blueprintsys.com/blueprint/api/v1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <Id>928</Id>
  <Name>Activity Type</Name>
  <DefaultValueIndex>1</DefaultValueIndex>
  <ValidValues>
    <ValidValue>Instant</ValidValue>
    <ValidValue>Responsive</ValidValue>
    <ValidValue>Waiting</ValidValue>
  </ValidValues>
</ChoicePropertyType>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/45921/metadata/choicePropertyTypes
```

Request Body

```
[
  {
    "Id": 928,
    "Name": "Activity Type",
    "DefaultValueIndex": "1",
```

```
"ValidValues": [  
  "Instant",  
  "Responsive",  
  "Waiting"  
]  
}  
]
```

Add Comment

Note: This request is supported in Blueprint 5.4 and later.

`api/v1/projects/{projectId}/artifacts/{artifactId}/comments`

DESCRIPTION Adds a single comment to the specified artifact.

URL STRUCTURE `[Blueprint_URI]/api/v1/projects/{projectId}/artifacts/{artifactId}/comments`

- **ProjectId:** The ID of the project containing the artifact that you want to add the comment to.
- **ArtifactId:** The ID of the artifact you want to add the comment to.

SUPPORTED METHODS ■ **POST**

REQUEST BODY The comment must be defined in the body of the request. Read more about [how to define a comment in the request body](#).

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments
```

Request Body

```
<?xml version="1.0" ?>  
<Comment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
```

```
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Description>Comment created by API</Description>
  <Status>Open</Status>
</Comment>
```

Response Body

```
<?xml version="1.0" ?>
<Comment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Author>
    <DisplayName>Admin</DisplayName>
    <Id>398</Id>
  </Author>
  <Description>&lt;br/><html&lt;br/>&lt;br/><head&lt;br/>&lt;br/><style
type="text/css"&lt;br/>&lt;br/>.c0 { font-size: 11px; line-height: 1.45000004768372 }
.c1 { margin: 0px; font-size: 11px; line-height: 1.45000004768372 }
&lt;br/></style&lt;br/>&lt;br/></head&lt;br/>&lt;br/><body
class="c0"&lt;br/>&lt;br/><p class="c1"&lt;br/>&lt;br/>&lt;br/><span
class="c0"&lt;br/>&lt;br/>Comment created by
API&lt;br/></span&lt;br/>&lt;br/></p&lt;br/>&lt;br/></body&lt;br/>&lt;br/></html&lt;br/>
&lt;br/></Description>
  <Id>71466</Id>
  <LastModified>2013-11-26T11:16:23.9633271-05:00</LastModified>
  <Version>0</Version>
  <IsClosed>>false</IsClosed>
  <Status>Open</Status>
</Comment>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/c
omments
```

Request Body

```
{
  "Status": "Open",
  "Description": "Comment created by API"
}
```

Response Body

```
{
  "LastModified": "2013-11-26T11:16:23.9633271-05:00",
  "IsClosed": false,
}
```



```
"Status": "Open",
  "Id": 71466,
  "Author": {
    "Type": "User",
    "Id": 398,
    "DisplayName": "Admin"
  },
  "Version": 0,
  "Description": "&lt;html&gt;&lt;head&gt;&lt;style type=\"text/css\"&gt;.c0 {
font-size: 11px; line-height: 1.45000004768372 } .c1 { margin: 0px; font-size:
11px; line-height: 1.45000004768372 } &lt;/style&gt;&lt;/head&gt;&lt;body
class=\"c0\"&gt;&lt;p class=\"c1\"&gt;&lt;span class=\"c0\"&gt;Comment created
by API&lt;/span&gt;&lt;/p&gt;&lt;/body&gt;&lt;/html&gt;"
}
```

Rate Comment

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/comments/  
{commentId}/feedback
```

DESCRIPTION Rates a comment by specifying whether you like or dislike the comment.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/
{artifactId}/comments/{commentId}/feedback

- **ProjectId:** The ID of the project containing the comment you want to rate.
- **ArtifactId:** The ID of the artifact containing the comment you want to rate.
- **CommentId:** The ID of the comment you want to rate. You can retrieve `CommentId` information via the [Get Artifact](#) request.

SUPPORTED METHODS ■ **POST**

URI PARAMETERS ■ **Like:** (*Required*) Specifies your feedback. You must specify one of the following values:

- **True:** Indicates that you agree with the comment and activates a thumbs-up icon.
- **False:** Indicates that you disagree with the comment and uses a thumbs-down icon.
- **None:** Resets the like status to neither *like* or *dislike*.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921/feedback?like=true
```

Response Body

```
<?xml version="1.0" ?>  
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">The comment  
was successfully rated.</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921/feedback?like=true
```

Response Body

```
"The comment was successfully rated."
```

Publish Comments

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/comments/publish
```

DESCRIPTION Publishes all unpublished comments associated with the specified artifact.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/{artifactId}/comments/publish

- **ProjectId:** The ID of the project containing the artifact with comments that you want to publish.
- **ArtifactId:** The ID of the artifact containing comments you want to publish.

SUPPORTED METHODS ■ **POST**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/publish
```

Response Header

```
{'content-length': '108', 'x-xss-protection': '1; mode=block', 'x-content-type-options': 'nosniff', 'expires': '-1', 'server': 'Blueprint', 'pragma': 'no-cache', 'cache-control': 'no-cache', 'date': 'Wed, 27 Nov 2013 17:30:59 GMT', 'x-frame-options': 'deny', 'content-type': 'application/xml; charset=utf-8'}
```

Response Body

```
<xml version="1.0">  
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">Comment  
successfully published.</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/  
comments/publish
```

Response Header

```
{'content-length': '33', 'x-xss-protection': '1; mode=block', 'x-content-type-  
options': 'nosniff', 'expires': '-1', 'server': 'Blueprint', 'pragma': 'no-  
cache', 'cache-control': 'no-cache', 'date': 'Wed, 27 Nov 2013 17:24:08 GMT',  
'x-frame-options': 'deny', 'content-type': 'application/json; charset=utf-8'}
```

Response Body

```
"Comment successfully published."
```

Update Comment

Note: This request is supported in Blueprint 5.4 and later.

```
/api/v1/projects/{projectId}/artifacts/{artifactId}/comments/{commentId}
```

DESCRIPTION Updates the specified comment.
You can only update a comment if Blueprint is configured to allow the updating of published comments and replies.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/{artifactId}/comments/{commentId}

- **ProjectId:** The ID of the project containing the comment you want to update.
- **ArtifactId:** The ID of the artifact containing the comment you want to update.
- **CommentId:** The ID of the comment you want to update.

SUPPORTED METHODS ■ **PATCH**

Note: You can [override the PATCH method](#) and use the [POST method](#) instead.

REQUEST BODY You must specify the new comment information in the body of the request. Read more about [how to update the comment in the request body](#).

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921
```

Request Body

```
<?xml version="1.0" ?>  
<Comment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
  xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <Description>Comment created by API</Description>  
  <Status>Open</Status>  
</Comment>
```

Response Body

```
<?xml version="1.0" ?>  
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">Comment  
Updated Successfully.</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/58392-1
```

Request Body

```
{  
  "Status": "Open",  
  "Description": "Comment created by API"  
}
```

Response Body

```
"Comment Updated Successfully."
```

Delete Comment

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/comments/{commentId}
```

DESCRIPTION Deletes the specified comment from an artifact.
You can only delete comments if Blueprint is configured to allow the deletion of published comments.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/{artifactId}/comments/{commentId}

- **ProjectId:** The ID of the project containing the comment you want to delete.
- **ArtifactId:** The ID of the artifact that contains the comment.
- **CommentId:** The ID of the comment you want to delete.

SUPPORTED METHODS

- **DELETE**

Note: You can [override the DELETE method](#) and use the [POST](#) method instead.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
http://today/api/v1/projects/1978492/artifacts/1978633/comments/377374
```

Response Header

```
{'content-length': '106', 'x-xss-protection': '1; mode=block', 'x-content-type-options': 'nosniff', 'expires': '-1', 'server': 'Blueprint', 'pragma': 'no-
```



```
cache', 'cache-control': 'no-cache', 'date': 'Wed, 27 Nov 2013 19:37:40 GMT',  
'x-frame-options': 'deny', 'content-type': 'application/xml; charset=utf-8'}
```

Response Body

```
<?xml version="1.0" ?>  
  <string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">Comment  
  successfully deleted.</string>
```

JSON Example

Request URI

```
http://today/api/v1/projects/1978492/artifacts/1978633/comments/377394
```

Response Header

```
{'content-length': '31', 'x-xss-protection': '1; mode=block', 'x-content-type-  
options': 'nosniff', 'expires': '-1', 'server': 'Blueprint', 'pragma': 'no-  
cache', 'cache-control': 'no-cache', 'date': 'Wed, 27 Nov 2013 18:51:38 GMT',  
'x-frame-options': 'deny', 'content-type': 'application/json; charset=utf-8'}
```

Response Body

```
"Comment successfully deleted."
```

Reply to Comment

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifact/{artifactId}/comments/  
{commentId}/replies
```

DESCRIPTION Reply to the specified comment.

URL [Blueprint_URI]/api/v1/projects/{projectId}/artifact/
STRUCTURE {artifactId}/comments/{commentId}/replies

- **ProjectId:** The ID of the project containing the comment you want to reply to.
- **ArtifactId:** The ID of the artifact containing the comment you want to reply to.
- **CommentId:** The ID of the comment you want to reply to.

SUPPORTED METHODS ■ **POST**

REQUEST BODY The reply must be defined in the body of the request. Read more about [how to specify a reply in the request body](#).

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/  
comments/583921/replies
```

Request Body

```
<?xml version="1.0" ?>
<Reply xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
<Description>&lt;html&gt;&lt;head&gt;&lt;/head&gt;&lt;body&gt;&lt;div&gt;&lt;p&
gt;&lt;span&gt;&lt;span&gt;Reply&lt;/span&gt;&lt;/span&gt;&lt;/p&gt;&lt;/div&gt
;&lt;/body&gt;&lt;/html&gt;</Description>
</Reply>
```

Response Body

```
<?xml version="1.0" ?>
<Reply xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Author>
    <DisplayName>Ivan Ivanov</DisplayName>
    <Id>1</Id>
  </Author>
  <Description>&lt;html&gt;&lt;head&gt;&lt;/head&gt;&lt;body&gt;&lt;div&gt;&lt;p&
gt;&lt;span&gt;&lt;span&gt;Reply&lt;/span&gt;&lt;/span&gt;&lt;/p&gt;&lt;/div&gt
;&lt;/body&gt;&lt;/html&gt;</Description>
  <Id>123</Id>
  <LastModified>2013-10-07T14:32:01.0000000</LastModified>
  <Version>0</Version>
  <CreatedOn>2013-10-07T14:32:01.0000000</CreatedOn>
</Reply>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/c
omments/583921/replies
```

Request Body

```
{
  "Description":
  "<html><head></head><body><div><p><span><span>Reply</span></span></p></div></bo
dy></html>"
}
```

Response Body

```
{
  "CreatedOn": "2013-10-07T14:32:01",
  "LastModified": "2013-10-07T14:32:01",
  "Id": 123,
  "Author": {
```

```
"Type": "User",
  "Id": 1,
  "DisplayName": "Fatima Ali"
},
"Version": 0,
"Description":
"<html><head></head><body><div><p><span><span>Reply</span></span></p></div></body></html>"
}
```

Rate Reply

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/comments/  
{commentId}/replies/{replyId}/feedback
```

DESCRIPTION Rates a reply by specifying whether you like or dislike the reply.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/
{artifactId}/comments/{commentId}/replies/
{replyId}/feedback

- **ProjectId:** The ID of the project containing the reply you want to rate.
- **ArtifactId:** The ID of the artifact containing the reply you want to rate.
- **CommentId:** The ID of the comment that is the parent of the reply you want to rate.
- **ReplyId:** The ID of the reply you want to rate.

SUPPORTED METHODS ■ **POST**

URI PARAMETERS ■ **Like:** (*Required*) Specifies your feedback. You must specify one of the following values:

- **True:** Indicates that you agree with the comment and activates a thumbs-up icon.
- **False:** Indicates that you disagree with the comment and activates a thumbs-down icon.
- **None:** Resets the like status to neither *like* or *dislike*.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921/replies/245444/feedback?like=true
```

Response Body

```
<?xml version="1.0" ?>  
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">The reply  
was successfully rated.</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921/replies/245444/feedback?like=true
```

Response Body

```
"The reply was successfully rated."
```

Update Reply

Note: This request is supported in Blueprint 5.4 and later.

```
/api/v1/projects/{projectId}/artifacts/{artifactId}/comments/  
{commentId}/replies/{replyId}
```

DESCRIPTION Updates the specified reply.
You can only update a reply if Blueprint is configured to allow the updating of published comments and replies.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/
{artifactId}/comments/{commentId}/replies/{replyId}

- **ProjectId:** The ID of the project that containing the reply you want to update.
- **ArtifactId:** The ID of the artifact containing the reply you want to update.
- **CommentId:** The ID of the comment associated with the reply you want to update.
- **ReplyId:** The ID of the reply you want to update.

SUPPORTED METHODS ■ **PATCH**

Note: You can [override the PATCH method](#) and use the [POST](#) method instead.

REQUEST BODY You must specify the new reply information in the body of the request. Read more about [how to update the reply in the request body](#).

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921/replies/233454
```

Request Body

```
<?xml version="1.0" ?>
<Reply xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">

<Description>&lt;html&gt;&lt;head&gt;&lt;/head&gt;&lt;body&gt;&lt;div&gt;&lt;p&
gt;&lt;span&gt;&lt;span&gt;Reply&lt;/span&gt;&lt;/span&gt;&lt;/p&gt;&lt;/div&gt;
&lt;/body&gt;&lt;/html&gt;</Description>
</Reply>
```

Response Body

```
<?xml version="1.0" ?>
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">The reply
was successfully updated.</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/comments/583921/replies/233454
```

Request Body

```
{
  "Description":
  "<html><head></head><body><div><p><span><span>Reply</span></span></p></div></body></html>"
}
```

Response Body

```
"The reply was successfully updated."
```


Delete Reply

Note: This request is supported in Blueprint 5.4 and later.

```
api/v1/projects/{projectId}/artifacts/{artifactId}/comments/  
{commentId}/replies/{replyId}
```

DESCRIPTION Deletes the specified reply from the specified artifact comment.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/artifacts/
{artifactId}/comments/{commentId}/replies/{replyId}

- ProjectId: The ID of the project containing the reply you want to delete.
- ArtifactId: The ID of the artifact containing the reply you want to delete.
- CommentId: The ID of the comment containing the reply you want to delete.
- ReplyId: The ID of the reply you want to delete.

SUPPORTED METHODS

- DELETE

Note: You can [override the DELETE method](#) and use the [POST](#) method instead.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/c  
omments/583921/replies/294840
```

Response Body

```
<?xml version="1.0" ?>  
<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">The reply  
was successfully deleted.</string>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/c  
omments/583921/replies/294840
```

Response Body

```
"The reply was successfully deleted."
```

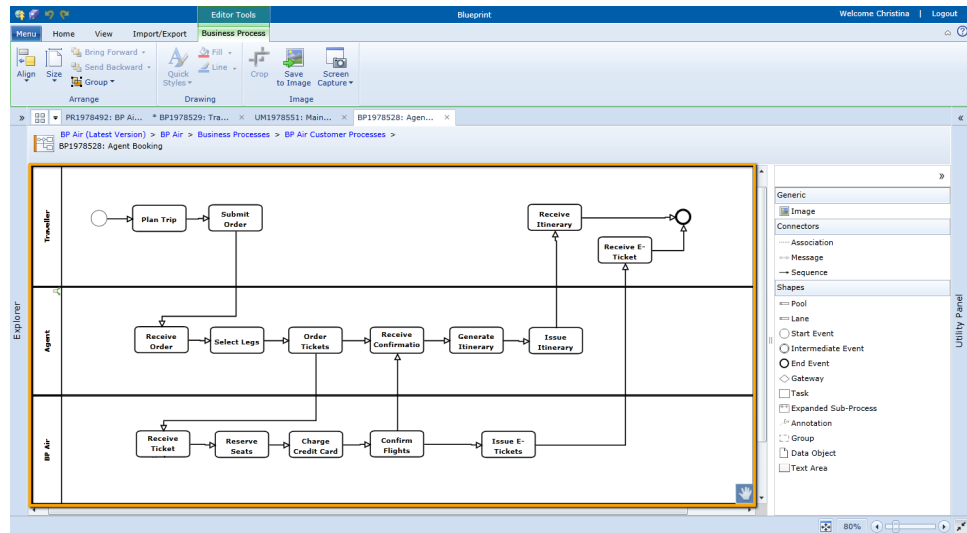
Get Artifact Image

Note: This request is supported in Blueprint 5.4 and later.

`api/v1/projects/{projectId}/artifacts/{artifactId}/image`

DESCRIPTION Retrieves the main image from the specified graphical artifact in PNG format. Graphical artifacts include use case diagrams, generic diagrams, business diagrams, domain diagrams, UI mockups, storyboards and use cases.

This is the main image from an example business process diagram:



URL STRUCTURE `[Blueprint_URI]/api/v1/projects/{projectId}/artifacts/{artifactId}/image`

- **ProjectId:** The ID of the project that contains the artifact image you want to get.
- **ArtifactId:** The ID of the artifact containing the image you want to get.

SUPPORTED METHODS

- **HEAD**
- **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/image
```

Response Body

The image data is contained in the response body.

The response header contains important information about the image, such as:

- `content-length`: The size of the response.
- `content-disposition`: The content type and the name of the file. For example: `image; filename="New Use Case 1.png"`.
- `content-type`: The response content type, which is `image/png` in this case.

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/220870/artifacts/220914/image
```

Response Body

The image data is contained in the response body.

The response header contains important information about the image, such as:

- `content-length`: The size of the response.
- `content-disposition`: The content type and the name of the file. For example: `image; filename="New Use Case 1.png"`.
- `content-type`: The response content type, which is `image/png` in this case.

Get Blueprint Product Version

Note: This request is supported in Blueprint 5.4 and later.

`api/info`

DESCRIPTION Retrieves the Blueprint product version.

URL [Blueprint_URI]/api/info
STRUCTURE

SUPPORTED METHODS ■ GET

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
https://production.blueprintcloud.com/api/info
```

Response Body

```
<?xml version="1.0" ?>  
<Info xmlns:i="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://www.blueprintsys.com/blueprint/api/v1">  
  <BlueprintServerVersion>5.4.0.250</BlueprintServerVersion>  
</Info>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/info
```

Response Body

```
{  
  "BlueprintServerVersion": "5.4"  
}
```

List Reviews

Note: This request is supported in Blueprint 5.5.1 and later.

```
api/v1/projects/{projectId}/reviews?baselineId={baselineId}&offset={offset}&limit={limit}
```

DESCRIPTION	<p>Retrieves a list of reviews by Project ID.</p> <p>Reviews are only included in the response if the user (that is was authenticated to obtain the token) has view permissions to those reviews.</p>
URL STRUCTURE	<p>[Blueprint_URI]api/v1/projects/{projectId}/reviews?baselineId={baselineId}&offset={offset}&limit={limit}</p> <ul style="list-style-type: none">■ ProjectId: The ID of the project that contains the reviews you want to retrieve.
SUPPORTED METHODS	<ul style="list-style-type: none">■ GET■ HEAD
REQUEST URI PARAMETERS	<ul style="list-style-type: none">■ BaselineId (optional): A specified baseline ID is used as a filter to display only formal reviews that are part of the baseline. By default, this parameter has no value and is not used (that is, all of the project reviews are returned).■ Offset (optional): Defines the paging offset (that is, index) at which the results start. Offset the default is 0.■ Limit (optional): Defines the number of items (projects, artifacts, etc) to retrieve per query. This parameter can be set to an integer. The default is 100. The minimum value is 1 and the maximum value is 500.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/864/reviews?baselineId=99
```

Response Body

```
<Reviews xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Review>
    <Id>3506300</Id>
    <Name>New Review 2</Name>
    <ProjectId>864</ProjectId>
    <BaselineId>99<BaselineId>
  </Review>
  <Review>
    <Id>3506302</Id>
    <Name>New Review 1</Name>
    <ProjectId>864</ProjectId>
  </Review>
</Reviews>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/864/reviews?baselineId=99
```

Response Body

```
[
  {
    "BaselineId":99,
    "Id":3506300,
    "Name":"New Review 2",
    "ProjectId":864
  },
  {
    "Id":3506302,
    "Name":"New Review 1",
    "ProjectId":864
  }
]
```


Get Review

Note: This request is supported in Blueprint 5.5.1 and later.

```
api/v1/projects/{projectId}/reviews/{artifactId}
```

DESCRIPTION Retrieves a review by Project ID and Artifact ID and returns information about the review.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{ProjectId}/reviews/{ArtifactId}

- **ProjectId:** The ID of the project that contains the review you want to retrieve.
- **ArtifactId:** The ID of the review artifact you want to retrieve.

SUPPORTED METHODS ■ **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/3473663/reviews/3506415
```

Response Body

```
<Review xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Id>3506415</Id>
  <Name>Milestone 1</Name>
  <ProjectId>3473663</ProjectId>
  <IsFormal>>false</IsFormal>
```

```
<ReviewStatus>Draft</ReviewStatus>
</Review>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/3473663/reviews/3506415
```

Response Body

```
{
  "IsFormal": false,
  "ReviewStatus": "Draft",
  "Id": 3506415,
  "Name": "Milestone 1",
  "ProjectId": 3473663
}
```

List ALM Targets

Note: This request is supported in Blueprint 5.5.1 and later.

`api/v1/projects/{projectId}/alm/targets`

DESCRIPTION	Retrieves a list of ALM targets. ALM targets are only included in the response if the user (that is was authenticated to obtain the token) has view permissions to those ALM targets.
URL STRUCTURE	[Blueprint_URI]/api/v1/projects/{projectId}/alm/targets <ul style="list-style-type: none">■ ProjectId: The ID of the project that contains the ALM targets you want to retrieve.
SUPPORTED METHODS	<ul style="list-style-type: none">■ GET■ HEAD

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets
```

Response Body

```
<AlmTargets xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <AlmTarget>
    <AlmType>HpAlmRest</AlmType>
    <BlueprintProjectId>1467764</BlueprintProjectId>
    <Domain>DEFAULT</Domain>
    <Id>40221</Id>
  </AlmTarget>
</AlmTargets>
```

```
<LastExportInfo>
  <BlueprintRevision>Live</BlueprintRevision>
  <QcExportFolderId>185028</QcExportFolderId>
  <QcExportPath>Requirements/CostcoTest</QcExportPath>
  <Timestamp>2014-07-11T17:48:24.1170083Z</Timestamp>
</LastExportInfo>
<Name>Hp Alm 11.5 Rest BlueprintTesting</Name>
<Project>BlueprintTesting</Project>
<Url>http://svmalml15:8080/qcbin/</Url>
</AlmTarget>
</AlmTargets>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets
```

Response Body

```
[
  {
    "Id":40221,
    "Name":"Hp Alm 11.5 Rest BlueprintTesting",
    "BlueprintProjectId":1467764,
    "AlmType":"HpAlmRest",
    "Url":"http://svmalml15:8080/qcbin/",
    "Domain":"DEFAULT",
    "Project":"BlueprintTesting",
    "LastExportInfo":{
      "Timestamp":"2014-07-11T17:48:24.1170083Z",
      "BlueprintRevision":"Live",
      "QcExportPath":"Requirements/CostcoTest",
      "QcExportFolderId":185028
    }
  }
]
```

Get ALM Job

Note: This request is supported in Blueprint 5.5.1 and later.

```
/api/v1/projects/{projectId}/alm/targets/{targetId}/jobs/{jobId}
```

DESCRIPTION Retrieves an ALM job by Project ID, Target ID and Job ID and returns information about the job.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/alm/targets/{targetId}/jobs/{jobId}

- **ProjectId:** The ID of the project that contains the ALM job you want to retrieve.
- **TargetId:** The ID of the ALM target for the job you want to retrieve.
- **JobId:** The ID of the ALM job you want to retrieve.

SUPPORTED METHODS

- **HEAD**
- **GET**

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Example

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets/40221/jobs/1705
```

Response Body

```
<JobInfo xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <IsJobMarkedForCancellation>>false</IsJobMarkedForCancellation>
  <JobEndDateTime>2014-07-11T17:51:38.7030000Z</JobEndDateTime>
```

```
<JobId>1705</JobId>
<JobOutput>
Started the first ALM export. Exported 7/7 artifact(s). Exported traces for 7/7
artifact(s). Finished ALM export.
</JobOutput>
<JobServerName>todayJobService@SVMTODAYWEB</JobServerName>
<JobStartDateTime>2014-07-11T17:50:41.180000Z</JobStartDateTime>
<JobType>HpAlmRestExport</JobType>
<Progress>100%</Progress>
<ProjectId>1467764</ProjectId>
<ProjectName>PR1467764: Test export traces to QC</ProjectName>
<Result i:type="ExportJobResult">
  <Records>
    <ExportJobResultRecord>
      <BlueprintArtifact>New Folder 13</BlueprintArtifact>
      <BlueprintId>PF1467766</BlueprintId>
      <Message>Successfully exported Blueprint item 1467766 by creating
corresponding Alm item 185029</Message>
    </ExportJobResultRecord>
    <ExportJobResultRecord>
      <BlueprintArtifact>New Textual Requirement 1</BlueprintArtifact>
      <BlueprintId>RQ1467776</BlueprintId>
      <Message>Successfully exported Blueprint item 1467776 by creating
corresponding Alm item 185030</Message>
    </ExportJobResultRecord>
    <ExportJobResultRecord>
      <BlueprintArtifact>New Textual Requirement 2</BlueprintArtifact>
      <BlueprintId>RQ1467771</BlueprintId>
      <Message>Successfully exported Blueprint item 1467771 by creating
corresponding Alm item 185031</Message>
    </ExportJobResultRecord>
    <ExportJobResultRecord>
      <BlueprintArtifact>New Use Case 1</BlueprintArtifact>
      <BlueprintId>UC1467772</BlueprintId>
      <Message>Successfully exported Blueprint item 1467772 by creating
corresponding Alm item 185032</Message>
    </ExportJobResultRecord>
    <ExportJobResultRecord>
      <BlueprintArtifact>New Use Case 2</BlueprintArtifact>
      <BlueprintId>UC1467767</BlueprintId>
      <Message>Successfully exported Blueprint item 1467767 by creating
corresponding Alm item 185033</Message>
    </ExportJobResultRecord>
    <ExportJobResultRecord>
      <BlueprintArtifact>New Actor 1</BlueprintArtifact>
      <BlueprintId>AC1467778</BlueprintId>
      <Message>Successfully exported Blueprint item 1467778 by creating
corresponding Alm item 185034</Message>
    </ExportJobResultRecord>
    <ExportJobResultRecord>
      <BlueprintArtifact>New UI Mockup 1</BlueprintArtifact>
      <BlueprintId>UM1467777</BlueprintId>
      <Message>Successfully exported Blueprint item 1467777 by creating
corresponding Alm item 185035</Message>
    </ExportJobResultRecord>
  </Records>
</Result>
<Status>Completed</Status>
<SubmittedDateTime>2014-07-11T17:50:36.527000Z</SubmittedDateTime>
<UserDisplayName>Default Instance Admin</UserDisplayName>
```

```
<UserId>1</UserId>
</JobInfo>
```

JSON Example

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets/40221
/jobs/1705
```

Response Body

```
{
  "SubmittedDateTime": "2014-07-11T17:50:36.5270000Z",
  "JobStartDateTime": "2014-07-11T17:50:41.1800000Z",
  "JobEndDateTime": "2014-07-11T17:51:38.7030000Z",
  "JobId": 1705,
  "Status": "Completed",
  "JobType": "HpAlmRestExport",
  "ProjectName": "PR1467764: Test export traces to QC",
  "UserId": 1,
  "UserDisplayName": "Default Instance Admin",
  "JobServerName": "todayJobService@SVMTODAYWEB",
  "Progress": "100%",
  "JobOutput": "Started the first ALM export.\nExported 7/7 artifact
(s).\nExported traces for 7/7 artifact(s).\nFinished ALM export.\n",
  "IsJobMarkedForCancellation": false,
  "ProjectId": 1467764,
  "Result": {
    "Records": [
      {
        "Type": "Success",
        "BlueprintId": "PF1467766",
        "BlueprintArtifact": "New Folder 13",
        "Message": "Successfully exported Blueprint item 1467766 by
creating corresponding Alm item 185029"
      },
      {
        "Type": "Success",
        "BlueprintId": "RQ1467776",
        "BlueprintArtifact": "New Textual Requirement 1",
        "Message": "Successfully exported Blueprint item 1467776 by
creating corresponding Alm item 185030"
      },
      {
        "Type": "Success",
        "BlueprintId": "RQ1467771",
        "BlueprintArtifact": "New Textual Requirement 2",
        "Message": "Successfully exported Blueprint item 1467771 by
creating corresponding Alm item 185031"
      },
      {
        "Type": "Success",
        "BlueprintId": "UC1467772",
        "BlueprintArtifact": "New Use Case 1",

```

```
        "Message": "Successfully exported Blueprint item 1467772 by
creating corresponding Alm item 185032"
    },
    {
        "Type": "Success",
        "BlueprintId": "UC1467767",
        "BlueprintArtifact": "New Use Case 2",
        "Message": "Successfully exported Blueprint item 1467767 by
creating corresponding Alm item 185033"
    },
    {
        "Type": "Success",
        "BlueprintId": "AC1467778",
        "BlueprintArtifact": "New Actor 1",
        "Message": "Successfully exported Blueprint item 1467778 by
creating corresponding Alm item 185034"
    },
    {
        "Type": "Success",
        "BlueprintId": "UM1467777",
        "BlueprintArtifact": "New UI Mockup 1",
        "Message": "Successfully exported Blueprint item 1467777 by
creating corresponding Alm item 185035"
    }
]
}
```


Add ALM Job

Note: This request is supported in Blueprint 5.5.1 and later.

```
api/v1/projects/{projectId}/alm/targets/{targetId}/jobs
```

DESCRIPTION Adds an ALM job to the specified project using the specified target.

URL STRUCTURE [Blueprint_URI]/api/v1/projects/{projectId}/alm/targets/{targetId}/jobs

- **ProjectId:** The ID of the project containing the artifacts that are being pushed as a part of the ALM job process.
- **TargetId:** The ID of the ALM target you want to use to create the ALM job.

SUPPORTED METHODS ■ **POST**

REQUEST BODY The new job must be defined in the body of the request. Read more about [how to define an ALM job](#) in the request body.

HTTP status codes

Refer to the section about [HTTP status codes](#).

Examples

XML Examples

ALM Export

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets/40221/jobs
```

Request Body

```
<AlmJobParameters xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <AlmJobType>AlmExport</AlmJobType>
  <JobParameters i:type="AlmExportParameters">
    <ConflictResolutionRule>Keep</ConflictResolutionRule>
    <ChangeSummaryJobId>1704</ChangeSummaryJobId>
  </JobParameters>
</AlmJobParameters>
```

Response Body

```
<JobInfo xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <IsJobMarkedForCancellation>>false</IsJobMarkedForCancellation>
  <JobEndTime>2014-04-25T16:34:21.0000000Z</JobEndTime>
  <JobId>1704</JobId>
  <JobOutput>2/2 artifacts completed.</JobOutput>
  <JobServerName>BlueprintService@DTTEST</JobServerName>
  <JobStartTime>2014-04-25T16:24:04.0000000Z</JobStartTime>
  <JobType>AlmExport</JobType>
  <Progress>100%</Progress>
  <ProjectId>1467764</ProjectId>
  <ProjectName>BP Air</ProjectName>
  <Result i:type="ExportJobResult">
    <Records>
      <ExportJobResultRecord>
        <BlueprintArtifact>Textual requirements</BlueprintArtifact>
        <BlueprintId>TR2</BlueprintId>
        <Message>no errors</Message>
      </ExportJobResultRecord>
      <ExportJobResultRecord>
        <BlueprintArtifact>graphical requirements</BlueprintArtifact>
        <BlueprintId>GR2</BlueprintId>
        <Message>failed to push</Message>
        <Type>Error</Type>
      </ExportJobResultRecord>
    </Records>
  </Result>
  <Status>Completed</Status>
  <SubmittedDateTime>2014-04-25T16:20:34.0000000Z</SubmittedDateTime>
  <UserDisplayName>Christina</UserDisplayName>
  <UserId>199</UserId>
</JobInfo>
```

Change Summary

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets/40221/
jobs
```

Request Body

```
<AlmJobParameters xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <AlmJobType>ChangeSummary</AlmJobType>
  <JobParameters i:type="ChangeSummaryParameters">
    <AlmRootPath>Enterprise Projects/BP Air</AlmRootPath>
    <AlmRootPathId>16</AlmRootPathId>
    <BaselineOrReviewId>3506300</BaselineOrReviewId>
    <IsFirstPush>true</IsFirstPush>
    <IsImageGenerationRequired>true</IsImageGenerationRequired>
  </JobParameters>
</AlmJobParameters>
```

Response Body

```
<JobInfo xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <IsJobMarkedForCancellation>>false</IsJobMarkedForCancellation>
  <JobId>1707</JobId>
  <JobOutput>Started collecting the change summary for the first ALM
export</JobOutput>
  <JobServerName>todayJobService@SVMTODAYWEB</JobServerName>
  <JobStartDateTime>2014-07-14T19:26:35.6400000Z</JobStartDateTime>
  <JobType>HpAlmRestChangeSummary</JobType>
  <Progress>0%</Progress>
  <ProjectId>1467764</ProjectId>
  <ProjectName>PR1467764: Test export traces to QC</ProjectName>
  <Status>Running</Status>
  <SubmittedDateTime>2014-07-14T19:26:33.6970000Z</SubmittedDateTime>
  <UserDisplayName>Christina</UserDisplayName>
  <UserId>199</UserId>
</JobInfo>
```

JSON Examples

ALM Export

Request URI

```
http://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets/40221/
jobs
```

Request Body

```
{
  "AlmJobType": "AlmExport",
  "JobParameters": {
    "Type": "AlmExportParameters",
    "ConflictResolutionRule": "Keep",
    "ChangeSummaryJobId": "1704"
  }
}
```

Response Body

```
{
  "SubmittedDateTime": "2014-04-25T16:20:34Z",
  "JobStartDateTime": "2014-04-25T16:24:04Z",
  "JobEndDateTime": "2014-04-25T16:34:21Z",
  "JobId": 1704,
  "Status": "Completed",
  "JobType": "AlmExport",
  "ProjectName": "BP Air",
  "UserId": 199,
  "UserDisplayName": "Christina",
  "JobServerName": "BlueprintService@DTTEST",
  "Progress": "100%",
  "JobOutput": "2/2 artifacts completed.",
  "IsJobMarkedForCancellation": false,
  "ProjectId": 1467764,
  "Result": {
    "Records": [
      {
        "Type": "Success",
        "BlueprintId": "TR2",
        "BlueprintArtifact": "Textual requirements",
        "Message": "no errors"
      },
      {
        "Type": "Error",
        "BlueprintId": "GR2",
        "BlueprintArtifact": "graphical requirements",
        "Message": "failed to push"
      }
    ]
  }
}
```

Change Summary

Request URI

```
https://production.blueprintcloud.com/api/v1/projects/1467764/alm/targets/40221/jobs
```

Request Body

```
{
  "AlmJobType": "ChangeSummary",
  "JobParameters": {
    "Type": "ChangeSummaryParameters",
    "IsImageGenerationRequired": false,
    "IsFirstPush": true,
    "BaselineOrReviewId": 3506300,
    "AlmRootPathId": 16,
    "AlmRootPath": "Enterprise Projects/BP Air"
  }
}
```

```
}  
}
```

Response Body

```
{  
  "SubmittedDateTime": "2014-07-14T19:45:20.1930000Z",  
  "JobStartDateTime": "2014-07-14T19:45:25.0300000Z",  
  "JobId": 1708,  
  "Status": "Running",  
  "JobType": "HpAlmRestChangeSummary",  
  "ProjectName": "PR1467764: Test export traces to QC",  
  "UserId": 199,  
  "UserDisplayName": "Christina",  
  "JobServerName": "todayJobService@SVMTODAYWEB",  
  "Progress": "1%",  
  "JobOutput": "Started collecting the change summary for the first ALM  
export.\n",  
  "IsJobMarkedForCancellation": false,  
  "ProjectId": 1467764  
}
```

Filter

Overview

The filter parameter allows you to narrow the results that are included in the response to your request. If no filter parameters are specified, all artifacts are returned. If a filter parameter is specified, artifacts are only returned if the artifact matches the specified criteria. The filter parameters can be used individually, or combined to narrow results further.

Syntax

The filter parameter must be specified using the following syntax:

```
filter={artifact-id eq ('1' or '2' or '3');artifact-type eq ('Actor');artifact-HasComments eq 'true';property-'My prop' eq 1;property-my2prop eq ('my name' or 'your name')}
```

Method

The filter parameter must be submitted using the [HTTP POST method](#).

To submit information using the HTTP POST method, you must add the following information to the request header to override the GET method.

```
X-HTTP-Method-Override: GET
```

Then, include your filter parameter in the request body. For example:

```
filter={artifact-type eq ('UI Mockup')};
```

Parameters

- **Artifact-Id**: If specified, artifacts are only returned if the ID of the artifact matches one of the given IDs.
- **Artifact-Type**: If specified, artifacts are only returned if the artifact type matches one of the given artifact types.
- **Artifact-HasAttachments**: If set to true, artifacts are only returned if there is at least one associated attachment. If set to false, artifacts are only returned if there are no associated attachments.
- **Artifact-HasComments**: If set to true, artifacts are only returned if there is at least one associated comment. If set to false, artifacts are only returned if there are no associated comments.
- **Property- [Name]**: If specified, artifacts are only returned if the artifact's property value matches the given value.

Supported operators

The supported operators varies depending on the data that you want to query. The table below outlines the supported operators for each type of data.

Query Type	eq (equal to)	eq (equal to)		bi (between inclusive ends)	gt (greater than)	lt (less than)	ge (greater than or equal to)	le (less than or equal to)
		AND	OR					
artifact-id	yes	n/a	yes	n/a	n/a	n/a	n/a	n/a
artifact-type	yes	n/a	yes	n/a	n/a	n/a	n/a	n/a
artifact-HasComments	yes	n/a	n/a	n/a	n/a	n/a	n/a	n/a
artifact-HasAttachments	yes	n/a	n/a	n/a	n/a	n/a	n/a	n/a
property-<plain text>	yes	n/a	yes	n/a	n/a	n/a	n/a	n/a
property-<rich text>	yes	n/a	n/a	n/a	n/a	n/a	n/a	n/a
property-<number>	yes	n/a	yes	no	no	no	no	no
property-<date>	yes	n/a	yes	yes	yes	yes	yes	yes
property-<choice>	yes	Yes	n/a	n/a	n/a	n/a	n/a	n/a
property-<user>	yes	Yes	n/a	n/a	n/a	n/a	n/a	n/a
property-'Created On'	no	n/a	n/a	Yes	Yes	Yes	Yes	Yes
property-'Last Edited On'	no	n/a	n/a	Yes	Yes	Yes	Yes	Yes
property-'Created By'	yes	n/a	n/a	n/a	n/a	n/a	n/a	n/a
property-'Last Edited By'	yes	n/a	n/a	n/a	n/a	n/a	n/a	n/a

Tips and Important Notes

- Filtering on users and groups:
 - When filtering on a user, you must use the user id prepended with the letter 'u'. For example, if the user ID was 20, you need to specify 'u20'.
 - When filtering on a group, you must use the group id prepended with the letter 'g'. For example, if the group ID was 18, you need to specify 'g18'.

- Filtering on dates and audit-datetime:
 - Dates are measured using just the date (example: '2013-05-11') while Author History fields ('Created On' and 'Last Edited On') are measured using nano-seconds (example: '2013-01-22T16:08:14.2170000Z'). Because of this difference, if you perform a greater than ('gt') filter, the results will be different. For example, a greater than operation on a date will return artifacts that match the following day. Alternatively, a greater than operation on the 'Created By' field will return artifacts after midnight on the same day.
- Filtering on custom properties:
 - If you are filtering on custom properties, you must include quotations around the property name if it includes a space. For example: property-'Focus Group'.

Examples

Retrieve all artifacts that have associated comments:

```
filter={artifact-HasComments eq ('true')};
```

Retrieve all artifacts of type 'Actor':

```
filter={artifact-type eq ('Actor')};
```

Retrieve all artifacts of either type 'Actor' or 'Use Case':

```
filter={artifact-type eq ('Use Case' or 'Actor')};
```

Retrieve all artifacts of type 'Textual Requirement' that have at least one attachment:

```
filter={artifact-type eq ('Textual Requirement'); artifact-HasAttachments eq ('true')};
```

Retrieve all artifacts where the Priority property of the artifact is set to 'High':

```
filter={property-priority eq ('High')};
```

Retrieve all artifacts where the Focus Group property of the artifact is equal to a group with an ID of 9 or 10:

```
filter={property-'Focus Group' eq ('g9' and 'u10')};
```

Retrieve all artifacts that were last edited between midnight on May 11th and midnight on May 14th inclusive:

```
filter={property-'Last Edited On' bi ('2013-05-11' and '2013-05-14')};
```


Retrieve all artifacts that were created after midnight on May 14th:

```
filter={property-'Created On' gt ('2013-05-14')}
```

Listing Artifacts in the Request Body

Overview

You must list the artifacts you are changing in the request message body when making any of the following requests:

- [Updating artifacts](#)
- [Publishing changes to artifacts](#)
- --Or--
- [Discarding changes to artifacts.](#)

Method

The list of artifacts must be submitted using the [HTTP POST method](#) or [HTTP PATCH method](#) depending on the type of request you are submitting.

Parameters

The following parameters can be specified within the `Artifact` element:

- `Id`: (Required.) Identifies the artifact numerically.
- `ArtifactTypeId`: Numerically identifies the type of artifact. The identifier can be obtained via the [Get Artifact Type](#) and [List Artifact Types](#) requests.
- `Name`: If you want to change the name of the artifact using the `UpdateArtifacts` call, you can specify a value.
- `ProjectId`: Identifies the project the artifact belongs to.
- Any required values for the property type of the artifact, [if you are specifying the property type](#).

Properties Element

Note: When adding a new artifact, you must specify values for all required properties.

If you want to specify a property type for your artifact, include a `Properties` parameter within the `Artifact` element.

Among others, the `Properties` element can contain the following elements:

- `Property`: (Required) The element containing the property specifications.
- `PropertyTypeId`: The corresponding elements, which can be populated in the [Get Artifact](#) call.

Examples

UpdateArtifacts Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>220914</Id>
    <Properties>
      <Property>
        <PropertyTypeId>9988</PropertyTypeId>
        <TextOrChoiceValue>Traveler Booking 1</TextOrChoiceValue>
      </Property>
    </Properties>
  </Artifact>
</Artifacts>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Id": 220914,
    "Properties": [
      {
        "PropertyTypeId": 9988,
        "TextOrChoiceValue": "Traveler Booking 1"
      }
    ]
  }
]
```

PublishArtifacts Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>220914</Id>
    <ProjectId>220870</ProjectId>
  </Artifact>
</Artifacts>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Id": "220914",
    "ProjectId": "220870"
  }
]
```

DiscardArtifacts Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>220914</Id>
    <ProjectId>220870</ProjectId>
  </Artifact>
</Artifacts>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Id": "220914",
    "ProjectId": "220870"
  }
]
```

Defining an Artifact in the Request Body

Overview

You must define the artifact in the request body when you are adding an artifact.

Note: You can only define one artifact in a single request.

Parameters

The following parameters must be specified within the `Artifact` element:

- `ArtifactTypeId`: Numerically identifies the type of artifact. The identifier can be obtained via the [Get Artifact Type](#) and [List Artifact Types](#) requests.
- `Name`: Identifies the name of the artifact.
- Any required properties of the applicable artifact type. Read more about the [Properties element](#).

Properties Element

If you want to specify property values for your artifact, include a `Properties` element within the `Artifact` element.

Note: If you are defining a property type for your artifact, you must specify values for required elements of the property type.

Among others, the `Properties` element can contain the following elements:

- `Property`: (Required.) The element of the property specifications.
- `PropertyTypeId`: The corresponding elements, which can be populated in the [Get Artifact](#) call.

Examples

AddArtifact Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Artifact xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <ArtifactTypeId>25</ArtifactTypeId>
  <Name>New Textual Requirement</Name>
  <ParentId>10</ParentId>
  <Properties>
    <Property>
```

```
<PropertyTypeId>235658</PropertyTypeId>
  <TextOrChoiceValue>Sample Description</TextOrChoiceValue>
</Property>
<Property>
  <NumberValue>15</NumberValue>
  <PropertyTypeId>235659</PropertyTypeId>
</Property>
</Properties>
</Artifact>
```

Here is an example of a request body in JSON format:

```
{
  "Name": "New Textual Requirement",
  "ParentId": 10,
  "ArtifactTypeId": 25,
  "Properties": [
    {
      "PropertyTypeId": 235658,
      "TextOrChoiceValue": "Sample Description"
    },
    {
      "PropertyTypeId": 235659,
      "NumberValue": 15.0
    }
  ]
}
```

Defining a Trace in the Request Body

Overview

The following services require you to define/specify a trace in the request body:

- [Add Traces](#)
- [Delete Traces](#)
- [Change Parent Trace](#)

In the request body you must specify:

- The other artifact in the trace relationship (for [Add Traces](#) and [Delete Traces](#) requests)
--Or--
- The new parent trace (for [Change Parent Trace](#))

Parameters

You must specify a `Traces` element first to contain any individual traces you want to specify by using the `Trace` element. Here is an example in XML:

```
<Traces>
  <Trace>
</Trace>
  <Trace>
</Trace>
</Traces>
```

The following parameters can be specified within each `<Trace>` element you are specifying:

- `ArtifactId`: The ID of the artifact whose trace relationship is being defined.
- `SubArtifactId`: The ID of the sub-artifact whose trace relationship is being defined.
- `ProjectId`: The ID of the project of the artifact whose trace relationship is being defined.
- `Type`: Defines the type of trace. The valid values are the following:
 - `Manual`: Only applicable when adding or deleting a trace.
 - `Parent`: Only applicable when moving an artifact.
- `Direction`: The direction of the trace. The valid values are the following:
 - `To`: From the artifact in the URI request to the artifact in the request body.
 - `From`: To the artifact in the URI request from the artifact in the request body.
 - `TwoWay`: The artifacts have a bi-directional trace.
- `IsSuspect`: Marks the trace as suspect.

Depending on the request, you are adding a trace relationship to, removing a relationship from, or establishing a new child-parent relationship with the artifact in the request body. In each context, some parameters are mandatory, while others are optional or ignored:

- **Adding a Trace:**
 - **Required:** `direction`, `type` (set to **Manual**), one of `artifactId` or `subArtifactId` (the other is optional)

- Optional: `projectId` and `isSuspect`
- Deleting a Trace:
 - Required: `type` (set to **Manual**), one of `artifactId` or `subArtifactId` (the other is optional), `direction` (if the query is using the `reconcileWithTwoWay` parameter, which is set to **true**)
 - Optional: `projectId`
 - Ignored: `isSuspect`
- Changing a Parent Trace:
 - Required: `type` (set to **Parent**), one of `artifactId` or `projectId`

Note: The artifact in the request body must belong to the same project as the artifact in the URI request.

- Ignored: All other properties are ignored.

Examples

AddTrace Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Trace>
    <ArtifactId>888020</ArtifactId>
    <Direction>To</Direction>
    <IsSuspect>>false</IsSuspect>
    <ProjectId>847739</ProjectId>
    <Type>Manual</Type>
  </Trace>
</Traces>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Type": "Manual",
    "ProjectId": 123456,
    "ArtifactId": 123,
    "IsSuspect": false
  },
  {
    "Type": "Manual",
    "ProjectId": 123456,
    "ArtifactId": 456,
    "IsSuspect": false
  },
  {
```



```
"Type": "Manual",
"ProjectId": 789,
"ArtifactId": 789,
"SubArtifactId": 789,
"IsSuspect": false
}
]
```

DeleteTraces Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Trace>
    <ArtifactId>843925</ArtifactId>
    <Type>Manual</Type>
  </Trace>
</Traces>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Type": "Manual",
    "ArtifactId": 847750
  }
]
```

ChangeParentTrace Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Traces xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Trace>
    <ArtifactId>999666</ArtifactId>
    <Type>Parent</Type>
  </Trace>
</Traces>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Type": "Parent",
```

```
    "ArtifactId": 999666  
  }  
]
```

Defining a User in the Request Body

Overview

You must define the user in the request body when you are creating or updating a user.

Note: You can only define one user in a single request.

Parameters

The following parameters must be specified within the `User` element:

- `Type*`: (Required) Identifies the type of artifact. Declare this as `""User""`.
- `Name*`: (Required) The username (and login name) of the user. This value is alphanumeric and must be between 4 and 255 characters.
- `DisplayName*`: (Required) The display name in the application that other users see.
- `Firstname`: The user's given name.
- `Lastname`: The user's surname.
- `GroupIds*`: Defines the groups in which the user is a member. Membership to a license group is required; membership to user groups is optional. Group identifiers can be obtained via the [List Groups](#) requests.
- `Department`: The organizational department of which the user is a member.
- `Title`: The user's job title.
- `Password`: The password used to log in to the application. This value must be between 8 and 128 characters, must contain a number, an upper-case letter, and must contain at least one non-alphanumeric character.
- `ExpiredPassword``: (true/false) Defines whether the user password expires. If the user password is configured to never expire, any password policy configured in the instance settings will not apply. Conversely, if passwords are configured in instance settings to never expire, any value configured here will not apply.
- `InstanceAdminRole``: Defines which Instance Administrator privileges the user has. By default no role is assigned.
- `Enabled``: Determines whether the user can log in to the application.
- `FallBack``: If federated authentication is enabled, determines whether the user is allowed to fall back from federated authentication.
- `Email`: The user's email address.
- Any required properties of the applicable artifact type. Read more about the [Properties element](#).

Properties Element

If you want to specify property values for your artifact, include a `Properties` element within the `Artifact` element.

Note: If you are defining a property type for your artifact, you must specify values for required elements of the property type.

Among others, the `Properties` element can contain the following elements:

- `Property`: (Required.) The element of the property specifications.
- `PropertyTypeId`: The corresponding elements, which can be populated in the `Get Artifact` call.

Examples

AddArtifact Example*****

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Users xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <User>
    <Email>dquade2084@dev.recall.com</Email>
    <Department>Devops</Department>
    <DisplayName>quaaade</DisplayName>
    <Firstname>Douglas</Firstname>
    <Lastname>Quade</Lastname>
    <Name>dquade</Name>
    <Enabled>>true</Enabled>
    <Password>@Quaaade#1</Password>
    <ExpiredPassword>>false</ExpiredPassword>
    <GroupIds
xmlns:i="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
      <i:int>8</i:int>
    </GroupIds>
    <Title>Engineer</Title>
  </User>
</Users>
```

Here is an example of a request body in JSON format:

```
[
  {
    "Type": "User",
    "Name": "dquade",
    "DisplayName": "quaaade",
    "Firstname": "Douglas",
    "Lastname": "Quade",
    "GroupIds": [
      "1",
      "8",
      "12"
    ],
    "Title": "Engineer",
    "Department": "Devops",
    "Password": "#fR33m4R5",
    "Enabled": true,
    "Email": "dquade2084@dev.recall.com"
  }
]
```

Defining a Comment in the Request Body

Overview

When adding or updating a comment, you must specify the comment text in the request body.

Method

To submit information using the HTTP POST method instead of HTTP PATCH, you must add the following information to the request header to override the PATCH method:

```
X-HTTP-Method-Override: PATCH
```

Parameters

The following parameters must be specified within the `Comment` element:

- **Description:** Write your comment in this element. Plain text formatting, HTML and inline styling are supported. If using HTML, the best practice is to use both `<html>` and `<body>` tags. External style sheets are not supported.
- **Status:** Identifies the status of the comment. For example: *open* or *closed*.

Note: Status values are configurable. You must specify a valid status value. You can obtain the status value of a comment via the [Get Artifact](#) request.

Examples

AddComment Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Comment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Description>Comment created by API</Description>
  <Status>Open</Status>
</Comment>
```

Here is an example of a request body in JSON format:

```
{
  "Status": "Open",
```

```
"Description": "Comment created by API"
}
```

UpdateComment Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Comment xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Description>Comment created by API</Description>
  <Status>Open</Status>
</Comment>
```

Here is an example of a request body in JSON format:

```
{
  "Status": "Open",
  "Description": "Comment created by API"
}
```

Defining a Reply in the Request Body

Overview

To [add](#) or [update](#) a reply to a comment, you must define your reply in the request body.

Method

The reply must be submitted using the [HTTP POST method](#) or [HTTP PATCH method](#) depending on the type of request you are submitting.

Parameters

The following parameters must be specified within the `Reply` element:

- **Description:** Write your reply in this element. Plain text formatting and HTML are supported.

Examples

ReplyToComment Example

Here is an example of a request body in XML format:

```
<?xml version="1.0" ?>
<Reply xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
<Description>&lt;html&gt;&lt;head&gt;&lt;/head&gt;&lt;body&gt;&lt;div&gt;&lt;p&
&gt;&lt;span&gt;&lt;span&gt;Reply&lt;/span&gt;&lt;/span&gt;&lt;/p&gt;&lt;/div&gt
&lt;/body&gt;&lt;/html&gt;</Description>
</Reply>
```

Here is an example of a request body in JSON format:

```
{
  "Description":
  "<html><head></head><body><div><p><span><span>Reply</span></span></p></div></bo
  dy></html>"
}
```

Defining an ALM Job in the Request Body

Overview

When you are making a request to create an ALM job, you must define the ALM job in the request body.

Two types of jobs can be defined:

- A push to an ALM target
- Or--
- A change summary for a push to an ALM target.

Initiating a push to an ALM target and initiating a change summary are separate ALM jobs that are a part of the same overarching action: efficiently exporting artifacts to an ALM system. Defining an ALM push sets up the export whereas defining a change summary involves choosing conflict resolutions for the export.

Note: You can only define one ALM job in a single request.

Method

The ALM job must be submitted using the [HTTP POST method](#).

Parameters

The following parameters must be specified within the `AlmJobParameters` element:

- `AlmJobType`: Identifies the type of ALM job. The ALM job value can either be `AlmExport` or `ChangeSummary`.
- `JobParameters`: If `AlmJobType` equals `ChangeSummary`, you must specify the `ChangeSummaryParameters` value in the `JobParameters` parameter. If `AlmJobType` equals `AlmExport`, you must specify the `AlmExportParameters` value in the `JobParameters` parameter. For more information about the `JobParameters` element, see the following section.

JobParameters Element

To specify property values for your ALM job, include the `JobParameters` element within the `AlmJobParameters` element. The `JobParameters` element can contain various parameters, including the following:

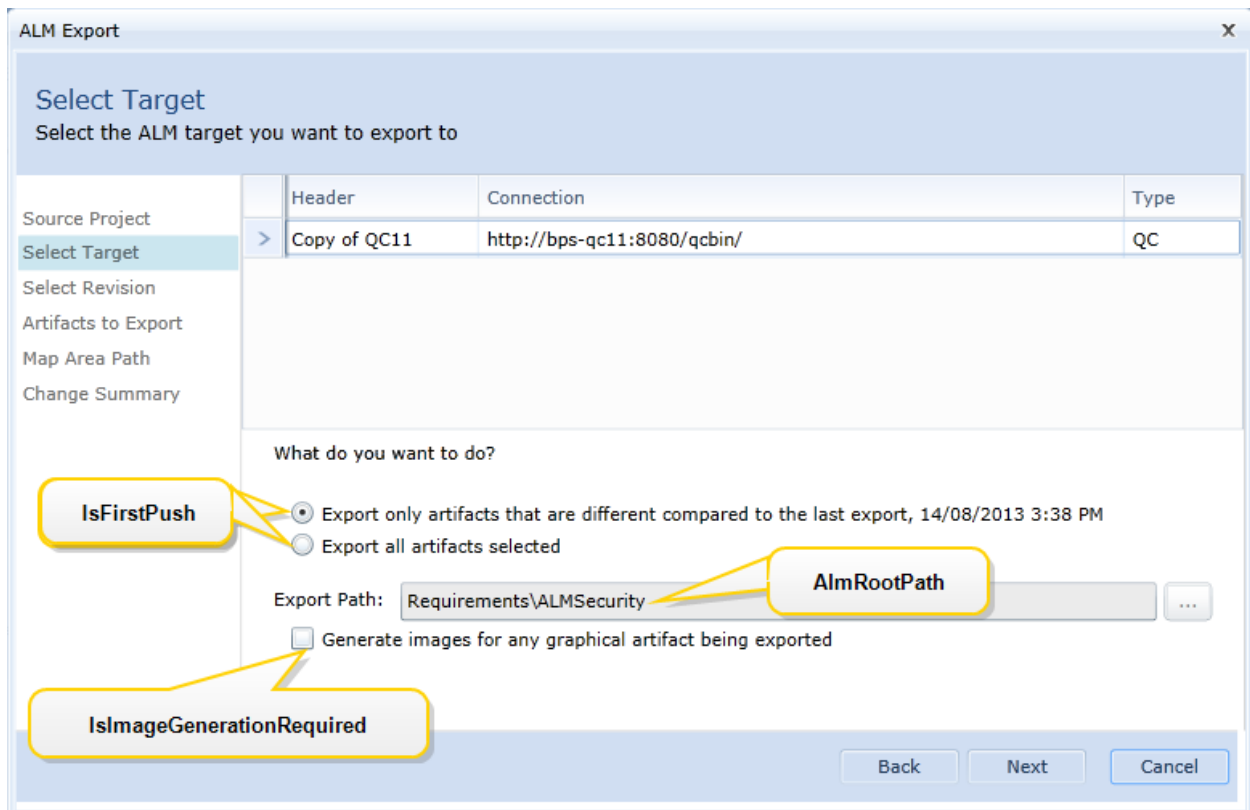
- `AlmRootPath`: Identifies the root path of the project folder which is used as a root for the ALM push. For example: `Enterprise Projects/BP Air`. (*Change summary jobs only.*)
- `AlmRootPathId`: This numerically identifies the folder that is used as a root for the ALM push. Although this parameter is optional, we recommend setting the value if you know ID of the target folder in QC/ALM system. (*Change summary jobs only.*)
- `BaselineOrReviewId`: Numerically identifies the baseline or review included in the ALM job.

(Required. Change summary jobs only.)

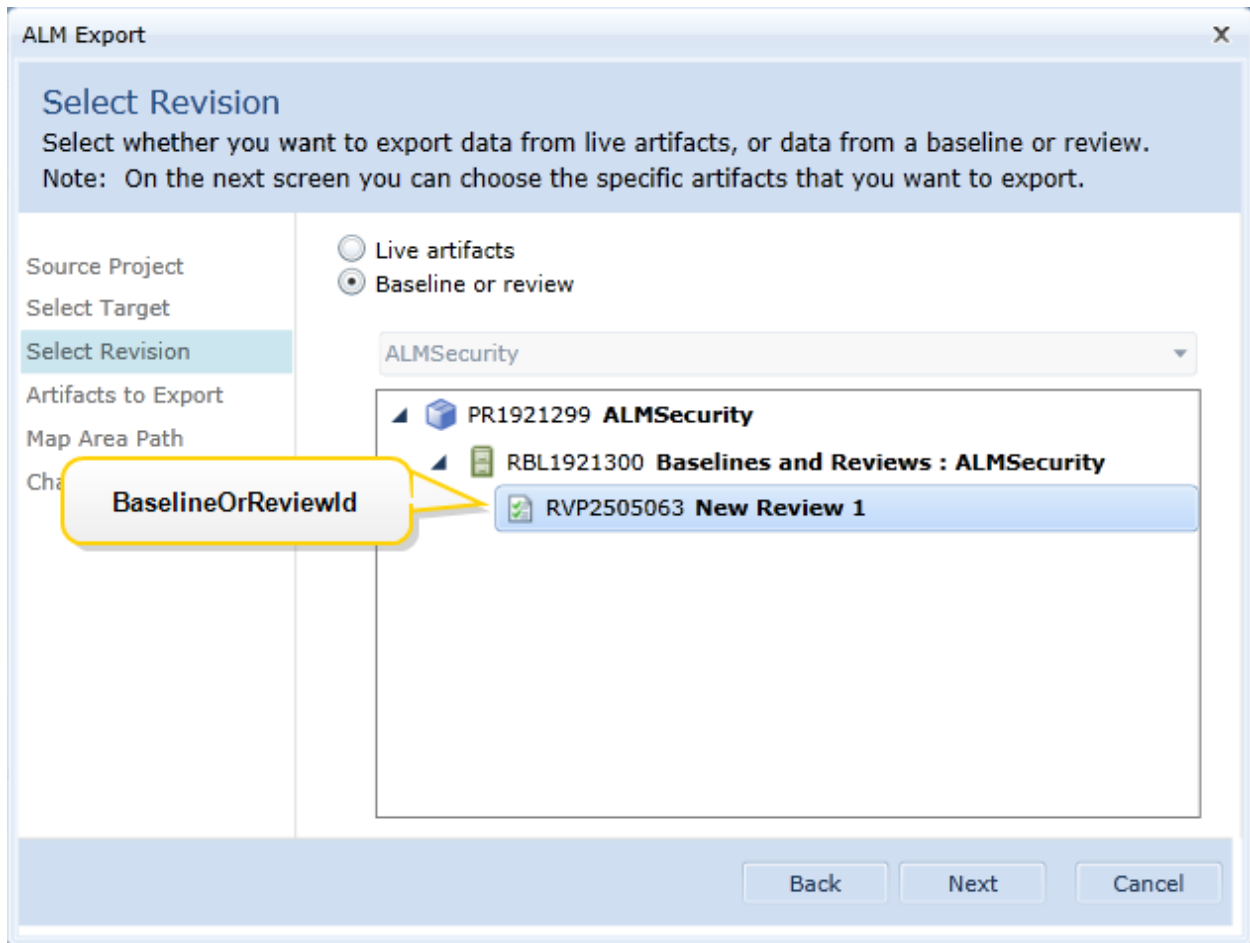
- **IsFirstPush**: A boolean (`true` or `false`) that identifies whether this is the first push to ALM or not. The default value is `true`. (Change summary jobs only.)
- **IsImageGenerationRequired**: A boolean (`true` or `false`) that allows you to specify whether you want images generated for any graphical artifacts in the export. The default value is `true`. (Change summary jobs only.)
- **ConflictResolutionRule**: Specifies how you want to resolve any conflicts that could arise when pushing artifacts to the ALM system. Specify the `Override` value if you want any artifacts you are pushing to override identical existing artifacts in the target system. Specify the `Keep` value if you do not want to override the artifacts in the target system. (Required. ALM export jobs only.)
- **ChangeSummaryJobId**: Numerically identifies the change summary job. (Required. ALM export jobs only.)

The following images show how particular `JobParameters` elements correspond to the *ALM Export* dialog within Blueprint.

Here is an image of the second screen that appears during an ALM export:



Here is an image of the third screen that appears during an ALM export:



Examples

AddALMJob Examples

ALM Export

Here is an example of a request body in XML format:

```
<AlmJobParameters xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <AlmJobType>AlmExport</AlmJobType>
  <JobParameters i:type="AlmExportParameters">
    <ConflictResolutionRule>Keep</ConflictResolutionRule>
    <ChangeSummaryJobId>1704</ChangeSummaryJobId>
  </JobParameters>
</AlmJobParameters>
```

Here is an example of a request body in JSON format:

```
{
  "AlmJobType": "AlmExport",
  "JobParameters": {
    "Type": "AlmExportParameters",
    "ConflictResolutionRule": "Keep",
    "ChangeSummaryJobId": "1704"
  }
}
```

Change Summary

Here is an example of a request body in XML format:

```
<AlmJobParameters xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <AlmJobType>ChangeSummary</AlmJobType>
  <JobParameters i:type="ChangeSummaryParameters">
    <AlmRootPath>Enterprise Projects/BP Air</AlmRootPath>
    <AlmRootPathId>16</AlmRootPathId>
    <BaselineOrReviewId>3506300</BaselineOrReviewId>
    <IsFirstPush>true</IsFirstPush>
    <IsImageGenerationRequired>true</IsImageGenerationRequired>
  </JobParameters>
</AlmJobParameters>
```

Here is an example of a request body in JSON format:

```
{
  "AlmJobType": "ChangeSummary",
  "JobParameters": {
    "Type": "ChangeSummaryParameters",
    "IsImageGenerationRequired": false,
    "IsFirstPush": true,
    "BaselineOrReviewId": 3506300,
    "AlmRootPathId": 16,
    "AlmRootPath": "Enterprise Projects/BP Air"
  }
}
```

Using the Content-Type: Multipart Method

Overview

You can use the Multipart method with the following requests:

- [Add Attachment](#)
- [Add Attachment to Subartifact](#)

To add data using the Multipart method, you must use boundary parameters for the data. For more information on using `MultipartContent`, see http://www.w3.org/Protocols/rfc1341/7_2_Multipart.html.

Example in C#

```
{
    //Call REST API
    var fetchRequest = (HttpWebRequest)WebRequest.Create(string.Concat(BaseURI,
"/", APICall));
    fetchRequest.Headers.Add("Authorization", "BlueprintToken" + " " +
authenticationToken);

    fetchRequest.Accept = "application/xml";
    fetchRequest.Method = "POST";
    fetchRequest.KeepAlive = true;

    fetchRequest.Credentials = System.Net.CredentialCache.DefaultCredentials;

    //Start creating multipart boundary
    string boundary = "-----" + DateTime.Now.Ticks.ToString
("x");
    byte[] boundarybytes = System.Text.Encoding.ASCII.GetBytes("\r\n--" +
boundary + "\r\n");

    fetchRequest.ContentType = "multipart/form-data; boundary=" + boundary;
    using (Stream rs = fetchRequest.GetRequestStream())
    {
        //Add starting boundary
        rs.Write(boundarybytes, 0, boundarybytes.Length);

        //Add Content headers (content disposition header)
        string headerTemplate = "Content-Disposition: form-data; name=\"{0}\";
filename=\"{1}\" \r\nContent-Type: {2} \r\n\r\n";
        string header = string.Format(headerTemplate, paramName, file, contentType);
        byte[] headerbytes = System.Text.Encoding.UTF8.GetBytes(header);
        rs.Write(headerbytes, 0, headerbytes.Length);

        //Add file
        using (var fileStream = new FileStream(file, FileMode.Open, FileAccess.Read))
        {
            var buffer = new byte[4096];
            int bytesRead = 0;
            while ((bytesRead = fileStream.Read(buffer, 0, buffer.Length)) != 0)
            {
                rs.Write(buffer, 0, bytesRead);
            }
        }
    }
}
```

```
    }  
  
    //Add last boundary  
    byte[] trailer = System.Text.Encoding.ASCII.GetBytes("\r\n--" + boundary + "--  
- \r\n");  
    rs.Write(trailer, 0, trailer.Length);  
    }  
    //Send request  
    var webResponse = fetchRequest.GetResponse() as HttpWebResponse;  
  
}
```

HEAD Method

The HTTP HEAD method lets you submit a request and receive a response with the response body omitted. The purpose of the HEAD method is to save bandwidth if you do not require the response data that is usually included in the body.

Example

If you want to determine the size of an attachment prior to downloading the file, you can use the HTTP HEAD method to submit your [Get Attachment](#) request. This allows you to determine the size of the attachment without having to download the file content.

Example

If you want to determine (approximately) how many artifacts exist for a particular filter, you can use the HTTP HEAD method to submit your [List Artifacts](#) request. This allows you to determine the size of the response without having to download the response body.

Python Example

```
def get_attachment():
    # obtain a token from our get_token() function
    token = get_token()

    # Of course, this ID would generally come from somewhere else instead of
    # hardcoding it.
    project_id = "220870"
    artifact_id = "220914"
    attachment_id = "3194"

    # build the resource uri for listing artifacts in a given project
    resource_uri = 'https://production.blueprintcloud.com/api/v1/projects/' +
    project_id + '/artifacts/' + artifact_id + '/attachments/' + attachment_id

    # specify header parameters
    request_header={
        'Authorization' : 'BlueprintToken ' + token,
        'Accept' : 'application/xml'
    }

    # specify the uri parameters
    uri_parameters_list={
    }

    # create the request uri with the uri parameters
    request_uri = resource_uri + '?' + urllib.parse.urlencode(uri_parameters_list)
    output_requesturi(request_uri)

    response = requests.head(request_uri, headers=request_header)
    return response.headers
```

GET Method

The HTTP GET method lets you submit a request with all request data included in the URI. The response includes the requested information in the body of the response.

Python Example

```
def list_projects():
    # obtain a token from our get_token() function
    token = get_token()

    # build the resource URI
    resource_uri = 'https://production.blueprintcloud.com/api/v1/projects'

    # build the request header
    request_header={
        'Authorization' : 'BlueprintToken ' + token,
        'Accept' : 'application/json'
    }

    # specify the URI parameters
    uri_parameters_list={
        'Location' : 'True'
    }

    # create the request uri
    request_uri = resource_uri + '?' + urllib.parse.urlencode(uri_parameters_list)
    output_requesturi(request_uri)

    # submit the request using HTTP GET method
    response = requests.get(request_uri, headers=request_header)
    return response
```

POST Method

The HTTP POST method allows you to submit a request with request data included in both the URI and the Body of the request. The response includes the requested information in the body of the response. If you are interested in using POST instead of PATCH or DELETE, read more about how you can [override the PATCH method](#) or [override the DELETE method](#) and use the POST method instead.

Method Override Example

For example, if you want to include the [Filter](#) parameter in your request, you must override the GET method and use POST instead.

To submit information using the HTTP POST method, you must add the following information to the request header to override the GET method.

```
X-HTTP-Method-Override: GET
```

Then, you can include request data in the request body. For example, the [Filter](#) parameter is always included in the request body using the HTTP POST method:

```
filter={artifact-type eq ('UI Mockup')};
```

Python Example

```
def list_artifacts():
    # obtain a token from our get_token() function
    token = get_token()

    # Of course, this ID would generally come from somewhere else instead of
    # hardcoding it.
    project_id = "220870"

    # build the resource uri for listing artifacts in a given project
    resource_uri = 'https://production.blueprintcloud.com/api/v1/projects/' +
    project_id + '/artifacts'

    # specify header parameters
    request_header={
        'Authorization' : 'BlueprintToken ' + token,
        'Accept' : 'application/json',
        'X-HTTP-Method-Override': 'GET'
    }

    # specify the uri parameters
    uri_parameters_list={
    }

    # create the request uri with the uri parameters
    request_uri = resource_uri + '?' + urllib.parse.urlencode(uri_parameters_list)
    output_requesturi(request_uri)
```



```
# create the request body
request_body = "filter={artifact-type eq 'Business Process Diagram'}"
output_requestbody(request_body)

# submit the request using HTTP POST method
response = requests.post(request_uri, request_body, headers=request_header)

return response
```

PATCH Method

Overview

The HTTP PATCH method allows you to update information based on data that you include in the body of the request.

For example, here is a list of artifacts included in the request body (in JSON format):

```
[
  {
    "Id": 220914,
    "Properties": [
      {
        "PropertyTypeId": 9988,
        "TextOrChoiceValue": "Traveler Booking 1"
      }
    ]
  }
]
```

Note: XML data in the request body must adhere to the [Blueprint REST API XML Schema](https://documentation.blueprintcloud.com/DeveloperCenter/RestAPI/Content/Schema/Blueprint_XMLSchema_RESTAPI_55.xsd) (https://documentation.blueprintcloud.com/DeveloperCenter/RestAPI/Content/Schema/Blueprint_XMLSchema_RESTAPI_55.xsd).

What should I do if I cannot use the PATCH method?

In some environments, the firewall may not support the PATCH method, or the PATCH method may be disabled. It is possible to use HTTP POST method instead of HTTP PATCH.

To submit information using the HTTP POST method instead of HTTP PATCH, you must add the following information to the request header to override the PATCH method:

```
X-HTTP-Method-Override: PATCH
```

Python Example

Below is a python example that submits an [Update Artifacts](#) request using the PATCH method.

```
def update_artifacts():
    # obtain a token from our get_token() function
    token = get_token()

    # Of course, this ID would generally come from somewhere else instead of
    # hardcoding it.
    project_id = "220870"

    # build the resource uri for listing artifacts in a given project
```

```
resource_uri = 'https://production.blueprintcloud.com/api/v1/projects/' +
project_id + '/artifacts/'

# specify header parameters
request_header={
    'Authorization' : 'BlueprintToken ' + token,
    'Accept' : 'application/json',
    'Content-Type' : 'text/json'
}

# specify the uri parameters
uri_parameters_list={
}

# create the request uri with the uri parameters
request_uri = resource_uri + '?' + urllib.parse.urlencode(uri_parameters_list)
output_requesturi(request_uri)

# the request body would generally be built from external data instead of
hardcoding it
# for demonstration purposes, an example is provided in both xml and json
format
artifactid_to_update = "220914"
propertyid_to_update = "9988"
new_property_value = "Traveler Booking 1"
xml_request_body = (''
<Artifacts xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.blueprintsys.com/blueprint/api/v1">
  <Artifact>
    <Id>%s</Id>
    <Properties>
      <Property>
        <PropertyTypeId>%s</PropertyTypeId>
        <TextOrChoiceValue>%s</TextOrChoiceValue>
      </Property>
    </Properties>
  </Artifact>
</Artifacts>
'')% (artifactid_to_update, propertyid_to_update, new_property_value)
json_request_body = (''
[
  {
    "Id": %s,
    "Properties": [
      {
        "PropertyTypeId": %s,
        "TextOrChoiceValue": "%s"
      }
    ]
  }
]
'')% (artifactid_to_update, propertyid_to_update, new_property_value)
# set the request body to the proper format for Content-Type:text/json
request_body=json_request_body
output_requestbody(request_body)

# submit the request using HTTP PATCH method
response = requests.patch(request_uri, request_body, headers=request_header)

return response
```

DELETE Method

The HTTP DELETE method lets you delete information based on request data included in the URI.

What should I do if I cannot use the DELETE method?

In some environments, the firewall may not support the DELETE method, or the DELETE method may be disabled. It is possible to use HTTP POST method instead of HTTP DELETE. To submit information using the HTTP POST method instead of HTTP DELETE, you must add the following information to the request header to override the POST method:

```
X-HTTP-Method-Override: DELETE
```

Python Example

```
def delete_artifact():
    # obtain a token from our get_token() function
    token = get_token()

    # Of course, this IDs would generally come from somewhere else instead of
    # hardcoding it.
    project_id = "115387"
    artifact_id = "115933"

    # build the resource uri for listing artifacts in a given project
    resource_uri = 'https://production.blueprintcloud.com/api/v1/projects/' +
    project_id + '/artifacts/' + artifact_id

    # specify header parameters
    request_header={
        'Authorization' : 'BlueprintToken ' + token,
        'Accept' : 'application/json',
        'X-HTTP-Method-Override': 'DELETE'
    }

    # create the request uri (add params if it's needed)
    request_uri = resource_uri

    # create the request body
    request_body = ""

    # submit the request using HTTP POST method
    response = requests.post(request_uri, request_body, headers=request_
    header)
    return response
```