

QC Catalogue API

1 Introduction

This document specifies an API for retrieving QC item definitions over HTTP, following a REST design pattern and using XML to represent data structures.

2 References

IETF RFC 5246, The Transport Layer Security (TLS) Protocol, Version 1.2

IETF RFC 7230, Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing

IETF RFC 7231, Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content

IETF RFC 7232, Hypertext Transfer Protocol (HTTP/1.1): Conditional Requests

IETF RFC 7234, Hypertext Transfer Protocol (HTTP/1.1): Caching

IETF RFC 5077, Transport Layer Security (TLS) Session Resumption without Server-Side State

W3C Recommendation, Extensible Markup Language (XML) 1.1 (Second Edition)

W3C Recommendation, XML Schema Definition Language (XSD) 1.1 Part 1: Structures

W3C Recommendation, Portable Network Graphics (PNG) Specification (Second Edition)

3 Common Provisions

3.1 HTTP

The server shall support the HTTP 1.1 protocol as specified in IETF RFC 7230, IETF RFC 7231, IETF RFC 7232 and IETF RFC 7234.

A persistent connection, as specified in Section 6.3 of IETF RFC 7230, should be maintained.

The server should implement TLS 1.2, as specified in IETF RFC 5246, for all endpoints.

The server should support session reuse via session identifiers as specified in IETF RFC 5246.

The server should support session reuse via session tickets as specified in IETF RFC 5077.

3.2 Base URI and API Version

Request URIs are specified relative to the following `{baseURI}`:

```
baseURI = hostPath apiVersion
```

`{hostPath}` is specific to each server.

`{apiVersion}` indicates the version of this specification to which the server conforms. It shall be equal to the following for this version of the specification:

```
apiVersion = "v1"
```

NOTE: The means by which clients are informed of the hostname and `{hostPath}` of a server is outside the scope of this document.

EXAMPLE: The following retrieves all items from the catalogue (see Section 4.1) from a server at localhost.

```
GET https://localhost/qccatalog/v1/items
```

3.3 XML Schemas

0 lists the XML Schema Definitions used to specify the contents of requests and response bodies.

Table 1 specifies the XML namespace and prefixes used by this specification.

Table 1. XML Namespaces and Prefixes.

Prefix	Namespace
qc	tag:ebu.ch,2017:ns:imf:qc:1.0

3.4 XML Documents

XML documents provided to and returned by endpoints shall be encoded using UTF-8 character encoding, as specified in W3C XML.

The `content-type` header for requests and responses containing a single XML document shall be "application/xml" with the `charset` parameter specified.

3.5 HTTP Responses

The following HTTP status codes may be returned. These status codes, unless specified otherwise, have the meaning specified in IETF RFC 7231.

- 301 Moved Permanently. Returned if the resource is no longer available at the request URI. The response shall contain a Location header containing the replacement request URI path.
- 401 Unauthorized. In addition to being returned as specified in IETF RFC 7231, i.e. returned if the Basic Authentication credentials are invalid, this status code is also returned if the client IP address is not authorized to access the resource.

- 404 Not Found. This status code is returned if an invalid `{apiVersion}` URI parameter is provided, i.e. an endpoint does not exist in the specified API version.
- 406 Not Acceptable. Returned if the server cannot provide a response that matches one of content types listed in the `Accept` request header.
- 405 Method Not Allowed. Returned if a request URI is not recognized by the server.
- 410 Gone. Returned if the resource is no longer available. The body of the response shall be empty.
- 415 Unsupported Media Type. Returned if the content type of the body, as indicated in `Content-Type` request header, is not supported by the server.
- 408 Request Timed Out
- 429 Too Many Requests
- 500 Internal Server Error
- 502 Bad Gateway
- 503 Service Unavailable
- 504 Gateway Timeout

3.6 URI Parameter Syntax

3.6.1 id

`{id}` shall conform to the syntax of the XML Schema type `qc:EBUQCIDType`.

3.6.2 version

`{version}` shall conform to the syntax of the XML Schema type `qc:EBUQCVersionType`.

3.7 XML Types

3.7.1 General

The following specifies semantics for XML Schema definitions used by the API.

3.7.2 ItemInfo

3.7.2.1 General

The `ItemInfo` element represents a version of a single item in the catalogue.

3.7.2.2 ThisVersion

The `ThisVersion` element contains the `EBUQCVersion` property of a published version of the item.

3.7.2.3 Versions

The `Versions` element lists the `EBUQCVersion` property of all published versions of the item.

3.7.2.4 Card



Figure 1. Pictorial Representation of a catalogue item.

The `Card` element provides the information necessary to generate a pictorial representation of the item, such as that illustrated in Figure 1.

The `Language` element specifies the language of the following items.

The `Title` element is a mnemonic for the item.

The `Line1` element is a brief description of the item.

The `Line2` element contains references associated with the item.

The `Line3` element summarizes pass criteria for the item.

The `Line4` element contains keywords associated with the information generated as output by the item.

The `Icon` element contains a list of alternative sources for an illustrative image associated with the item.

The `FullImage` element contains a list of alternative sources for a complete localized pictorial representation of the item, such as that illustrated in Figure 1.

For both the `Icon` and `FullImage` elements, the `Location` element is an absolute URI for the image resource, and the `Width` and `Height` elements are the width and height of the image, respectively. The image resource conforms to W3C Portable Network Graphics (PNG) Specification.

3.7.2.5 Item

The `Item` element contains the version of the item specified in `ThisVersion`.

4 Requests

4.1 GetItems

4.1.1 Description

This request retrieves a list of all versions of all items in the catalogue.

To minimize the impact of repeated polling by clients, the server should implement the entity-tag feature specified in IETF RFC 7232 for this request.

When making this request, a client:

- should not make regular requests less than 24 hours apart; and
- shall support the entity-tag feature field as specified in IETF RFC 7232.

4.1.2 Method and Target

```
GET {uri-base}/items
```

4.1.3 Body

No body shall be present.

4.1.4 Responses

4.1.4.1 200 OK

The response shall consist of a single XML Document with an instance of the `Items` element as its root.

Each `/Items/ItemInfo` element shall correspond to the latest published version of an item in the catalogue, and `Items` shall include all items of the catalogue.

4.2 GetLatestItem

4.2.1 General

This request retrieves the latest published version of an item from the catalogue.

This request should be used if the `EBUQCID` of an item is known but its version is not.

The `GetItems` request should be used instead if retrieving the latest version of significantly more than one item is needed.

EXAMPLE: The following retrieves the item whose EBUQCID is equal to "0002B".

```
GET https://localhost/qccatalog/v1/items/0002B
```

4.2.2 Method and Target

```
GET {uri-base}/items/{id}
```

4.2.3 Body

No body shall be present.

4.2.4 Responses

4.2.4.1 200 OK

The response shall consist of a single XML Document with an instance of the `ItemInfo` element as its root.

The `ItemInfo` element shall correspond to the highest published version of the item whose EBUQCID is equal to `{id}`.

4.2.4.2 404 Not Found

This response shall be returned if there is no published item whose EBUQCID is equal to `{id}`.

4.3 GetItemVersion

4.3.1 Description

This request retrieves a specific version of an item from the catalogue.

Since an item with a given combination of `EBUQCVersion` and `EBUQCID` is immutable, the caller should cache the results of a request for a specific version of the item, and no further request for the same specific version of the item should be made.

4.3.2 Method and Target

```
GET {uri-base}/items/{id}/versions/{version}
```

4.3.3 Body

No body shall be present.

4.3.4 Responses

4.3.4.1 200 OK

The response shall consist of a single XML Document with an instance of the `ItemInfo` element as its root

The `ItemInfo` element shall correspond to the published item whose `EBUQCVersion` corresponds to `{version}` and whose `EBUQCID` is equal to `{id}`.

4.3.4.2 404 Not Found

This response shall be returned if there is no published item whose `EBUQCVersion` corresponds to `{version}` and whose `EBUQCID` is equal to `{id}`.

Annex A XML Schema Definitions

The following elements contain the XML Schema Definitions used herein:

- qc-catalogue-api-schema.xsd
- qc.xsd