Entitlement API Specification
Scenarios

Version
2.02 (Oct 2020)

Audience
All consumers of the getFTR Entitlement API

Contributors
getFTR Technical Steering Group (TSG)

Member	Publisher
Gareth Wright	John Wiley
Mike Petras	American Chemical Society (ACS)
Artem Artemyev	Springer Nature
David Greenwald	Elsevier
Paul Smith	Taylor and Francis
Donatas Bacius	Equal Experts
Chris Shillum	GetFTR

Document Revision History

Version	Date	Comment	Author(s)
1.0	Oct 24 2019	First version	TSG
2.0	July 2020	Second version	TSG
2.01	Aug 2020	Correction to Scenario #8. Removed accessType	TSG
2.02	Oct 2020	Clarified other Digital Object Types and support. Extended Scenario #3. Added Scenarios: #12, #13, #14 and #15.	TSG
2.1	July 2021	Added permFree to #5	TSG

Table of Contents

Scenarios

Note, all content URLs shown in responses are purely for example purposes and their structure and format is outside the scope of this specification. They are at the discretion of the Publisher.

For brevity and clarity URLs ending *"?.."* imply request parameters are expected to support authentication where appropriate (i.e. "Smart Links")

#1 Entitled + VoR + IP

A single DOI and multiple Org IDs are provided in the request. The IP is matched with a subscribing organisation who is entitled to the VoR.

Request

```
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "yes",
            "accessType": "paid",
            "org": {
                "ipv4": "1.2.3.4"
            },
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc"
                },
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/full/123.abc"
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#2 Entitled + VoR + EntityID

A single DOI and multiple Org IDs are provided in the request. The entityID IP is matched with a subscribing organisation who is entitled to the VoR.

Request

```json
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "yes",
            "accessType": "paid",
            "org": {
                "entityID": "https://idp.example.org"
            },
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc?.."
                },
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/full/123.abc?.."
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#3 Maybe Entitled

"Maybe Entitled" scenario can arise when either:

- Authenticating via SAML and the Institution/Department cannot be uniquely identified based on the Org identifiers in the request. Deferred authentication can give rise to this case.
- Authenticating via SAML **or** IP and utilising Metered access, where there is a limited volume of "access tokens".
- When requesting entitlement to a container level object (e.g. a Book) and the Organisation has entitlements to some of the items within the container (e.g. Book Chapters). This scenario is supported at the discretion of the Publisher based on their business model.

A VoR is returned with a "maybe" entitled status:

Request

```
{
    "org": {
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "maybe",
            "accessType": "paid",
            "org": {
                "entityID": "https://idp.example.org"
            },
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc?.."
                },
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/full/123.abc?.."
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

See Scenario [#15](#15) for a Book example.

#4 Entitled + SAML attributes

The Organisation can now be uniquely identified as additional SAML attributes have been provided. The entityID and attribute combination uniquely matches a subscribing organisation who is entitled to the VoR. This pattern applies to "eduPersonScopedAffiliation" as well.

Request

```json
{
    "org": {
        "entityID": "https://example.idp.org",
        "openAthensOrgID": "999"
    },
    "dois": [
        "123.abc"
    ]
}
```

<u>Response</u>

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "yes",
            "accessType": "paid",
            "org": {
                "entityID": "https://idp.example.org",
                "openAthensOrgID": "999"
            },
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc?.."
                },
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/full/123.abc?.."
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#5 Open Access / Free / Perm Free

A single DOI and multiple Organisation identifiers are provided in the request. The document is Open Access / Free / Perm Free therefore none of the Organisation identifiers are required to establish entitlement, and are not returned. Anyone is entitled to the VoR.

<u>Request</u>

```
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "yes",
            "accessType": "open",
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc"
                },
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/full/123.abc"
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#6 Entitled + Non authenticating Org ID

A single DOI and a non-authenticating Org ID is provided (e.g. Ringgold, GRID, Ror). The Organisation can be mapped to an entityID and access provided to the VoR.

Request

```
{
    "org": {
        "ringgoldID": "777"
    },
    "dois": [
        "123.abc"
    ]
}
```

This scenario is to future proof the API specification, ready for when alternative institutional IDs are broadly adopted across the GetFTR participants.

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "yes",
            "accessType": "paid",
            "org": {
                "ringgoldID": "777"
            },
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc?.."
                },
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/full/123.abc?.."
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#7 Unentitled + Known Org + AV

A single DOI and multiple Org IDs are provided in the request. The entityID is matched to a subscribing institution, the organisation is not entitled to the VoR but is granted an Alternate Version (Publisher discretion).

Request

```
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "no",
            "org": {
                "entityID": "https://idp.example.org"
            },
            "av": [
                {
                    "contentType": "application/epub+zip",
                    "url": "https://publisher.org/doi/av/123.abc?.."
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#8 Unentitled + Known Org + No AV

A single DOI and multiple Org IDs are provided in the request. The entityID is matched to a subscribing institution, the organisation is not entitled to the VoR and is either not granted an Alternate Version or one is not available (Publisher discretion).

Request

```
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "no",
            "org": {
                "entityID": "https://idp.example.org"
            },
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

#9 Unentitled + Known Org

A single DOI and multiple Org IDs are provided in the request. The entityID was matched to a subscribing institution, therefore it is returned in the response. But, the Organisation isn't entitled to the document.

Request

```json
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "no",
            "org": {
                "entityID": "https://idp.example.org"
            },
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

It is at the discretion of the Publisher to return an Alternate Version in this scenario.

#10 Unentitled + Unknown Org

A single DOI and multiple Org IDs are provided in the request. None of the identifiers can be matched to a subscribing organisation or entitlement, therefore unentitled and "org" is not in the response.

Request

```json
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "no",
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```

It is at the discretion of the Publisher to return an Alternate Version in this scenario.

#11 Non 200 status codes

A number of scenarios can arise where a non 200 response can be returned in the response body. For example an unknown DOI.

```json
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "999.bad"
    ]
}
```

All responses with a non 200 will look like:

```json
{
    "entitlements": [
        {
            "doi": "...",
            "statusCode": ...
        }
    ]
}
```

For an exhaustive list of statusCodes refer to the API specification appendix.

#12 Supported Digital Object Types (Specification)

The API specification has been designed to support any Digital Object that can be identified by a DOI and accessed via a URL. Examples of Digital Object types:

- Book (ToC)
- Book Chapter
- Journal Article
- ...

All of the outlined scenarios are equally applicable to all types.

It is important to note that not all Publishers will support all theoretically possible Digital Object types, see Unsupported Digital Objects Types.

#13 Unsupported Digital Object Types (Publisher)

When a request for an unsupported Digital Object type is made against a Publisher's Entitlement API, the response should always be a 404, inline with Scenario 11.

Examples of Digital Object types:

- Book (ToC)
- Book Chapter
- Journal Article
- ...

Semantically, the Entitlement resource cannot be found for this Digital Object, hence 404.

An unknown DOI or an unknown entitlement have the same response outcome, both 404.

Request

```json
{
    "org": {
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc.ch1"
    ]
}
```

Response

The response should always be a 404, inline with Scenario 11.

```json
{
    "entitlements": [
        {
            "doi": "123.abc.ch1",
            "statusCode": 404
        }
    ]
}
```

#14 Entitled + Book Chapter + VoR + IP

Same as Scenario #1, with the example framed around a Book Chapter for clarity and completeness.

A single DOI (Book Chapter) and multiple Org IDs are provided in the request. The IP is matched with a subscribing organisation who is entitled to the Chapter's VoR.

Request

```json
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc.ch1"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc.ch1",
            "statusCode": 200,
            "entitled": "yes",
            "accessType": "paid",
            "org": {
                "ipv4": "1.2.3.4"
            },
            "vor": [
                {
                    "contentType": "application/pdf",
                    "url": "https://publisher.org/doi/pdf/123.abc.ch1"
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc.ch1"
        }
    ]
}
```

#15 Maybe Entitled + Book + VoR + IP

Same as Scenario [#3](#), with the example framed around a Book for clarity and completeness.

A single DOI (Book) and multiple Org IDs are provided in the request. The IP is matched with a subscribing organisation who is entitled to <u>some</u> of the chapters within the Book, and the Publisher has decided that a "maybe" response fits their business model.

<u>Request</u>

```
{
    "org": {
        "ipv4": "1.2.3.4",
        "entityID": "https://example.idp.org"
    },
    "dois": [
        "123.abc"
    ]
}
```

Response

```json
{
    "entitlements": [
        {
            "doi": "123.abc",
            "statusCode": 200,
            "entitled": "maybe",
            "accessType": "paid",
            "org": {
                "ipv4": "1.2.3.4"
            },
            "vor": [
                {
                    "contentType": "text/html",
                    "url": "https://publisher.org/doi/book/123.abc"
                }
            ],
            "document": "https://example.publisher.com/doi/abs/123.abc"
        }
    ]
}
```