



Europäisches
Patentamt
European
Patent Office
Office européen
des brevets

Patent Information

Open Patent Services RESTful Web Services

Reference Guide

Version 1.3.19

Revision history

Date	Version	Description
01/03/2011	1.0.0	Content development; master draft consultation version
16/05/2011	1.0.1	Update according to changes in the new OPS release
20/06/2011	1.0.2	Update according to changes in the next OPS release
12/12/2011	1.0.3	Update to Family priority-claim response
01/03/2012	1.0.4	Update according to 3.0 OPS release
19/11/2012	1.1	Updated to reflect CPC changes
30/05/2013	1.2	Updated for OPS 3.1 and user Registration
10/10/2013	1.2.8	Corrections to some Registered user response behaviour
08/11/2013	1.2.9	Number Service status codes update
18/11/2013	1.2.10	Timezone specified
16/04/2014	1.2.11	Quota error messages updated
05/09/2014	1.2.12	Fixed links. Improved information related to image services.
15/09/2014	1.2.13	CQL queries are updated
28/01/2015	1.2.14	Corrected the table with valid combinations of endpoints. CA authority supports fulltext.
24/04/2015	1.2.15	Register CQL specification link is corrected
15/01/2016	1.2.16	Added date formats available in CQL. Added extended CQL section. Minor CQL fixes in relation to restrictions and syntax. Added "cpcc" index.
25/01/2016	1.2.17	Removed <ops:meta elapsed-time="...">. Added INPADOC family total-result-count.
16/02/2016	1.2.18	Added new error message RequestTimeout.
15/04/2016	1.3.0	Register UPP information
10/06/2016	1.3.1	Reworked constituents table. Updated CQL examples according to user's request.
22/05/2017	1.3.2	Hyperlinks update and encryption protocols
07/08/2017	1.3.3	Update the epodoc publication format
26/09/2017	1.3.4	Removed all UPP and anonymous access references. Updated description of developer's area.
04/12/2017	1.3.5	Updated all the mentions of OPS version 3.1 in URLs to the version 3.2. Included some addition information for developer's area. Added table with preBRE codes.
15/01/2018	1.3.6	Fixed inaccuracy in a value of header Content-Type.
04/05/2018	1.3.7	Removed information about weekly free quota for registered users.
03/08/2018	1.3.8	Updated authentication response view.
25/01/2019	1.3.9	OPS errors are always in XML format
07/05/2019	1.3.10	Updated fulltext collection.
27/05/2019	1.3.11	Added CPC-I.
04/07/2019	1.3.12	Updated CPCI documentation
09/07/2019	1.3.13	Updated fair use link
24/07/2019	1.3.14	Added example with proximity search and relational operators for CPCI
05/09/2019	1.3.15	Updated the fulltext country code list
14/09/2020	1.3.16	Correction to countries list and searching Range limits

16/09/2021	1.3.17	Correction of the text in “Range control” article
24/05/2022	1.3.18	Unitary Patent support
31/05/2023	1.3.19	OPS Register with UPP constituent search identifiers

TABLE OF CONTENTS

1.	INTRODUCTION.....	6
1.1.	What is OPS?.....	6
1.2.	Patent information relevant to OPS.....	6
1.3.	EP Patent lifecycle and reference types.....	8
2.	OPS CONCEPTS.....	9
2.1.	Input	9
2.1.1.	Request structure.....	9
2.1.2.	Input format.....	11
2.1.3.	Rules for constructing the input patterns.....	12
2.2.	Output	15
2.2.1.	Response structure.....	15
2.2.2.	Common response structures (XML).....	19
2.2.3.	Error messages.....	30
2.3.	Registration & OPS Fair use policy	31
2.3.1.	User Registration for access.....	32
2.3.2.	Authentication & Access Token handling.....	33
2.3.3.	Dynamic fair use monitoring.....	40
2.3.4.	Data usage API.....	44
2.3.5.	Encryption protocols.....	47
3.	OPS SERVICES	48
3.1.	PUBLISHED-DATA SERVICES	48
3.1.1.	Bibliographic data.....	51
3.1.2.	Fulltext inquiry and retrieval including description or claims.....	61
3.1.3.	Images inquiry and retrieval.....	67
3.1.4.	Equivalents only or with abstract, biblio, full-cycle, images.....	73
3.2.	Family service	82
3.2.1.	Family data (without constituents).....	85
3.2.2.	Family data with <code>biblio</code>	86
3.2.3.	Family data with <code>legal</code>	87
3.2.4.	Family data with <code>biblio</code> and <code>legal</code>	89
3.3.	Number-service	91
3.3.1.	From <code>docdb</code> to <code>epodoc</code>	98

3.3.2.	From original to docdb (JPO example)	98
3.3.3.	From docdb to original (JPO example)	99
3.3.4.	From original to epodoc (USPTO example)	100
3.3.5.	From epodoc to original	101
3.3.6.	From original to docdb (PCT example)	101
3.4.	Register service	103
3.4.1.	Register retrieval	103
3.4.2.	Register search	113
3.5.	Legal service	117
3.6.	Classification services	121
3.6.1.	CPC Retrieval	122
3.6.2.	CPC Media retrieval	128
3.6.3.	CPC Search	129
3.6.4.	Classification Mapping service	132
3.7.	CPC-I	134
3.7.1.	Combination-sets	136
3.7.2.	CPCI optional condensed format	139
4.	APPENDIX	141
4.1.	Relevant acronyms and abbreviations	141
4.2.	CQL index catalogue	142
4.3.	The epodoc publication format	148
4.4.	Expressing PCT application numbers in docdb format	149
4.5.	Number-service Use Cases	149
4.6.	Further reading	154

1. INTRODUCTION

- 1.1. What is OPS?
- 1.2. Patent information relevant to OPS
- 1.3. EP Patent lifecycle and reference types

[top of document](#)

1.1. What is OPS?

[Open Patent Services](#) (or OPS) provides web services for machine-to-machine queries that deliver production stable patent data from the European Patent Office (EPO). OPS services are free of charge and available 24 hours a day, 7 days a week. Please read the [fair use charter](#) for details about using OPS.

Getting started

Beginning with version 3.0, OPS implements all services with a REST-style architecture.

This reference guide aims to provide the information and relevant details you need for automated retrievals of raw patent data using OPS RESTful services.

In [OPS Services](#) section, the request examples are real and can be copied and pasted into the browser where you can see the OPS responses. **Note**, the request response examples will appear in the browser stylised with XSL, however all responses can be viewed as XML by using 'View source'.

Note, for the pagination of this document some graphics have been scaled down. To see them in greater detail, please use the zoom function of your document reader. Thus, if you print this you may still need to refer to it electronically.

1.2. Patent information relevant to OPS

In the table below you will find the patent fundamentals relevant to OPS concepts.

Concept	Description
Patent application	The formal "paperwork" filed by an applicant, (or by a patent attorney or patent agent on the applicant's behalf) seeking to obtain a patent for a specific invention. There are several parts to a patent application, for example the description and the claims.
Patent publication	The first patent publication is often the published patent application , 18 months after a priority date. Other publications are typically the patent specification or in some cases the search report.
Patent priority	The Paris Convention for the Protection of Industrial Property

Concept	Description
	of 1883, established the system of priority rights, under which applicants have up to 12 months from first filing of their patent application (usually in their own country) to submit further or subsequent applications in each signatory country and claim the original priority date. The first filing data (country, number and date) is thus known as priority data for that invention. For further details, especially for details about priority number formats please refer to the priority number (on Espacenet.com).
Patent publication kind code	A code which includes 1 or 2 letters and in many cases a number, used to distinguish the kind of published patent document. For example, the publication of an application for a patent with or without a search report, and the level of the publication, for example first publication, second publication or corrected publication. For more details see the kind codes (on Espacenet.com).
Patent publication date	The date when a described invention becomes publicly available. Further details publication date (on Espacenet.com).
Patent application claims	The part of the patent that defines the scope of the legal protection sought for the invention. For further details, see claims (on Espacenet.com).
Patent citation(s)	A patent document cited. Citations are not only added by the patent applicant but also by the examiners of the patent application. Patents citations may be added during the different steps of the granting process (search report, examination, third party observations, opposition) and thus added to the patent data.
Simple patent family	All documents sharing exactly the same set of priorities. For further details see patent families (on www.epo.org).
Patent family	All documents sharing directly or indirectly at least one priority.

Table 1

1.3. EP Patent lifecycle and reference types

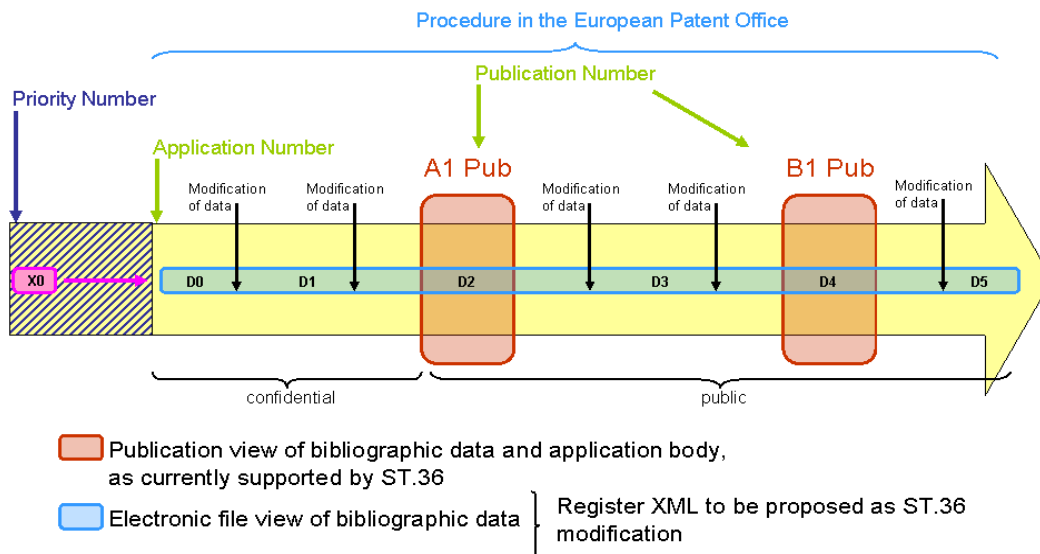


Figure 1

Diagram summary:

- Each document can be referred to with a priority, application and publication number.
- Information (X0) (as shown on the striped box on the left) from the priority document could serve as the basis for the initial patent information (D0) (shown as the first part of the bibliographic stream of data in the lifecycle.)
- Every modification (bibliographic data, description, claims ...) results in a subsequent set of documents that represent the state of the file of the patent application. The register provides a view on the modifications related to the bibliographic data.
- The bibliographic data in a published document represents a snapshot of the bibliographic data at time of publication, i.e. the set of valid bibliographic/register data that is part of the publication.
- Register data is the public view of the electronic file when the application enters the public phase, i.e. after the first publication in the European Patent office.

2. OPS CONCEPTS

2.1. Input

- 2.1.1. Request structure
- 2.1.2. Input format
- 2.1.3. Rules for constructing the input patterns

2.2. Output

- 2.2.1. Response structure
- 2.2.2. Common response structures (XML)
- 2.2.3. Error messages

2.3. Registration & OPS Fair use policy

- 2.3.1. User Registration for access
- 2.3.2. Authentication & Access Token handling
- 2.3.3. Dynamic fair use monitoring
- 2.3.4. Data usage API
- 2.3.5. Encryption protocols

[top of document](#)

2.1. Input

The following sub-sections focus on the patterns used to construct requests, including the elements used specifically by OPS requests and their definitions.

2.1.1. Request structure

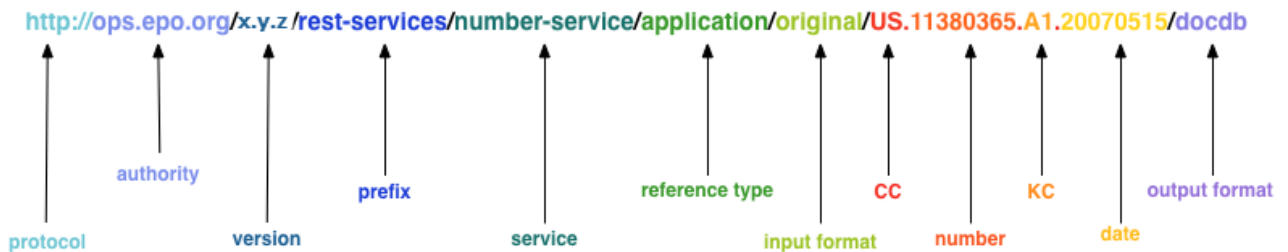
How the generic OPS request URI is constructed:

```
protocol/authority/[version]/prefix/service/  
reference-type/input-  
format/input/[endpoint]/[constituent(s)]/output-format
```

Note, URI parts in **bold** are mandatory, square brackets [] signify optional parts.

The `output-format` is used only by the [Number-service \(section 3.3\)](#).

Request structure concept



Note, [Classification service \(section 3.6\)](#) does not use the same request structure (as shown above.) You will find the request structure and examples in that section.

POST method

In addition to GET requests, OPS also supports the HTTP POST method. This is especially helpful if you want to request a large number of publications via the bibliographic data bulk retrieval (see [section 3.1, Published-data services](#)). It is also an alternative to using brackets in the URI when dealing with numbers in original number formats including special characters like slashes and dots (see [section 3.3, Number-service](#)).

In most cases, in order to use the POST method, the request URI is the same as with GET but without the `input`:

```
protocol/authority/prefix/service/reference-type/input-format/[endpoint]/[constituent(s)]
```

Note, URI parts in square brackets [] are optional.

The `input` would then go into the request body. When posting multiple numbers, you can provide them on separate lines or in one-line separated with commas.

Although most of the tools set `Content-Type` automatically, OPS has a strict regulation on the value of `Content-Type` HTTP request header. You will need to make sure that this header is not empty and the value for it is `text/plain` for any OPS POST request.

```
Content-Type: text/plain
```

For POST requests example please refer to the specific service description.

Tip: While GET requests can easily be tested using a browser, this is not possible for POST requests. You can use the [Mozilla Firefox RESTClient Add-On](#) to test all kinds of HTTP methods as well as HTTP headers.

Request concept elements

URI part	Description
protocol	Usually http
authority	Usually it's the host name. The OPS production authority is <code>ops.epo.org</code>
version	Currently, 3.0.
prefix	Always rest-services to distinguish the use of the OPS RESTful services
service	Service name; see OPS services (section 3) .
reference-type	There are 3 types: publication, application, priority – all of which are mandatory for requests.
input-format	Mandatory for requests; format-specific; see below Input format (section 2.1.2) .
input:	Consists of 4 input parts: CC, number, [KC] and [date]. Note, in Classification service (section 3.6) the input is structured differently.
CC	The country code; mandatory part of the input; format-specific; see below Input format (section 2.1.2) .
number	Number that belongs to the reference type input; mandatory part of the input; format-specific; see below Input format (section 2.1.2) .
[KC]	Kind code; <i>optional part of the epodoc input.</i> (Note, the kind code can be replaced by the asterisk * wildcard.) See Rules for constructing the input patterns (section 2.1.3) .
[date]	Date; in YYYYMMDD format only; <i>optional part of the input.</i>
[endpoint]	<i>Optional URI part;</i> Used by published-data services and provides various parts of published data. Mandatory, for Number-service (section 3.3) .
[constituent(s)]	<i>Optional part(s) of the URI.</i> A response modifier. Several can be included in a request. They are separated with commas.

Table 2

2.1.2. Input format

The original number format is the domestic numbering format of each patent office. These number formats vary and are documented in the [WIPO standard ST10c](#). **Note,** the original format is used only by the OPS [number-service \(section 3.3\)](#).

The docdb format is derived from the original number formats. The docdb format is related to the EPO's "DOCDB" database, a global patent database, which stores the domestic number formats in a consistent way. For more information on PCT numbers, please refer to the [Appendix section 4.4 – Expressing PCT application numbers in docdb format](#).

For search purposes, the EPO has created the "EPODOC" database, which has introduced a strict number normalization or formatting, especially for application and priority numbers. Thus it is known as the `epodoc` format.

A publication number in `epodoc` format refers to the initial publication in the patent lifecycle (as shown in Figure 1 in [section 1.3](#)). Characteristic of an `epodoc` number is it may have a letter combined into it at the end, **e.g.** NL1016699C.

For additional details, please see in the [Appendix section 4.3 – The epodoc publication format](#).

Reference type/format	Example that appears on the actual document
application/original	MD a 2005 0130
application/docdb	MD.20050130.A
application/epodoc	MD20050000130

Table 3

2.1.3. Rules for constructing the input patterns

There are **3 input formats**: (of which each has a specific pattern):
`original`, `docdb`, `epodoc`.

original

Input consists of 4 possible parts:

- *country code* (CC) - *optional*
- **number** (unformatted string) - **mandatory**
- *kind code* (KC) - *optional*
- *date* (date) - *optional*

Note, the date format used in OPS is ALWAYS YYYYMMDD.

Note:

- `original` format is **USED ONLY** by the OPS [number-service \(section 3.3\)](#).

docdb

Input consists of 4 possible parts:

- **country code** (CC) - **mandatory**
- **number** (number) - **mandatory**
- **kind code** (KC) - **mandatory**
- *date* (date) - *optional*

Note, the date format used in OPS is ALWAYS YYYYMMDD.

epodoc

Input consists of 3 possible parts:

- **number** (the epodoc number string) - **mandatory**
- *kind code* (KC) - *optional* docdb kind code
- *date* (date) - *optional*

Note, the date format used in OPS is ALWAYS YYYYMMDD.

Note:

- in certain countries (e.g. EP, WO, FR, GB) a publication number in epodoc format refers to the initial publication in the patent lifecycle (as shown in [section 1.3](#)); you can add a corresponding docdb kind code to refer to another publication of the patent lifecycle.

Input construction rule 1: concatenation with dots

Example of original **input, reference-type** application:

US. (08/921,321) .19970829 [CC.number.date]

Note:

- see input handling with brackets below.

Example of docdb **input, reference-type** application:

US. 92132197.A.19970829 [CC.number.KC.date]

Example of epodoc **input, reference-type** application:

US19970921321.19970829 [number.date]

Input construction rule 2: handling special characters with brackets

- Some patent numbers may include slashes (/), dots (.) and commas (,) e.g. **US08/921,321** or **CH 99947655.9**. To use numbers like these (that have special characters) for an OPS number-service request, **the commas must to be enclosed inside brackets ()**. Otherwise the service will understand these commas as separators. Thus, input for requests should become: **(US08/921,321)** and **CH.(99947655.9)**.
- These number types could be used as input if reserved and unsafe characters are encoded (see '[encoding](#)' below.) Although **OPS DOES NOT ACCEPT any encoding of the slash**, commas (,) are acceptable when encoded with %2C. Making the number example look like this:
US08%2C921,321

Input construction rule 3: encoding

For various reasons some characters present the *possibility* of being misunderstood within URIs. These characters should also always be encoded when they are used for a different from its intended purpose as URL section separators. **Note**, the first 4 in the list, **in bold are mandatory** encoding for OPS. The others are highly recommended.

URL reserved and unsafe characters in HTTP

Name	Character	Code
Question mark	?	%3F
'At' symbol	@	%40
'Hash' or "pound" symbol	#	%23
Percentage symbol	%	%25
Dollar	\$	%24
Ampersand	&	%26
Plus	+	%2B
Comma	,	%2C
Colon	:	%3A
Semi-colon	;	%3B
Equals	=	%3D
A empty space in the line, string or URI.	<i>SPACE</i>	%20
Double-quotation mark	"	%22
'Less than' or left-angled bracket	<	%3C
'Greater than' or right-angled bracket	>	%3E
Left curly brace	{	%7B
Right curly brace	}	%7D
Vertical bar or "pipe"		%7C
Caret	^	%5E
Tilde	~	%7E
Left square bracket	[%5B
Right square bracket]	%5D
Grave accent	`	%60

Table 4

Note, in OPS these characters SHOULD NEVER BE ENCODED:

Forward slash	/
Back slash	\

Table 5

Format	Examples	Description
original	DE 20 2007 016 308.8 OPS compliant presentation is with brackets (see above): DE.(20 2007 016 308.8) And with the <i>space</i> escaped this number becomes: DE.(20%202007%20016 308.8)	The original application number , as found in the original patent publication.
docdb	DE.202007016308.U	Application number with country code (DE) and kind code (U). Note, all parts must be separated with dots [.]
epodoc	DE200720016308U	Application number with country code (DE), number (200720016308) and kind code (U) are usually combined (no dots).

Table 6

Input construction rule 4: functional modifications

Commas

– For OPS requests, commas can be used only to separate constituents that are used by the [published-data service \(section 3.1\)](#), the [family service \(section 3.2\)](#) and [register services \(section 3.4\)](#).

Note, these constituents are explained in their respective sections.

– Commas are also used in bulk requests to separate entities.

2.2. Output

The following sub-sections focus on the various types of XML output in OPS responses as well as the XML structures used by the OPS services.

2.2.1. Response structure

When using OPS RESTful services it is important to know how the data is structured to ensure that you can properly request and retrieve what you need.

By default almost all responses are in XML and are styled with XSL for a more visually expressive output if you view the response using a web browser. However you can always see the XML behind these responses by using the "view source" function in your browser.

All XML responses are encapsulated within the `ops:world-patent-data` element, but their structure greatly depends on the web service type. This section contains elements, which are important parts of a service response.

The schema below is the basis upon which all the other schemas are referenced.



Figure 2

Aside from the OPS XML responses (encapsulated by `ops:world-patent-data`) there are other response types that can be obtained as part of your request.

HTTP response content types

OPS service	Possible value for Accept header
Published-data	application/fulltext+xml
Published-data	application/exchange+xml
Published-data/images	image/pdf
Published-data/images	application/tiff
Published-data/images	image/png
Number-service	application/ops+xml

OPS service	Possible value for Accept header
Classification/cpc	application/cpc+xml
CPC/images	image/gif
Register	application/register+xml
ALL OPS SERVICES that support XML	application/json
ALL OPS SERVICES that support XML	application/javascript

Table 7

You will find more information on what the respective responses look like in the [OPS services chapters](#).

JSON

As shown in Table 7 all OPS services that support XML are also capable of giving a JSON response. JSON is a lightweight, language-independent, text-based format for interchanging data and is often used in AJAX requests and responses. For more detailed information on the JSON format, please refer to <http://www.json.org/>.

Requesting JSON instead of XML is simple. Just change the Accept header in your GET- or POST-Request to `application/json` and use the same request URI as usual.

All elements in the XML response are also present when requesting JSON.

For the transformation of the output from XML to JSON, OPS follows the so-called *BadgerFish convention*. This convention dictates the following 12 rules for the conversion:

- Element names become object properties
- Text content of elements goes in the \$ property of an object.
- Nested elements become nested properties
- Multiple elements at the same level become array elements.
- Attributes go in properties whose names begin with @.
- Active namespaces for an element go in the element's @xmlns property.
- The default namespace URI goes in @xmlns.\$.
- Other namespaces go in other properties of @xmlns.
- Elements with namespace prefixes become object properties, too.

Example Request:

```
GET http://ops.epo.org/rest-services/published-data/publication/epodoc/EP1000000.A1/biblio
```

```
Accept: application/json
```

Response:

```

{"ops:world-patent-data": {
  "@xmlns": {
    "ops": "http://\./ops.epo.org",
    "$": "http://\./www.epo.org/exchange",
    "ccd": "http://\./www.epo.org/ccd",
    "xlink": "http://\./www.w3.org/1999/xlink"
  },
  "exchange-documents": {"exchange-document": {
    "@system": "ops.epo.org",
    "@family-id": "19768124",
    "@country": "EP",
    "@doc-number": "1000000",
    "@kind": "A1",
    "bibliographic-data": {
      "publication-reference": {"document-id": [
        {
          "@document-id-type": "docdb",
          "country": {"$": "EP"},
          "doc-number": {"$": "1000000"},
          "kind": {"$": "A1"},
          "date": {"$": "20000517"}
        },
        {
          "@document-id-type": "epodoc",
          "doc-number": {"$": "EP1000000"},
          "date": {"$": "20000517"}
        }
      ]},
      // remaining values removed for readability
    } } }
}

```

Note, for a quick test using the JSON format in your browser you can add `.json` to the end of your OPS URI, e.g.:

<http://ops.epo.org/rest-services/published-data/publication/epodoc/EP1000000.A1/biblio.json>

Overview of XML namespaces

OPS provides information in XML, embedded in a structure containing different namespaces, which are used in other EPO systems. Depending on the web service you call, the response might contain elements from several namespaces.

Data Structure document	Namespace	Description
Open Patent Services input and output schema	ops	This is the main namespace that references all of the others. Defines the structure of the XML output

Data Structure document	Namespace	Description
		documents in OPS v.3.0. Additionally, it defines a structure used for exchanging legal information about the patent publication.
EPO Exchange Document	exch	Defines structures for exchanging patent documents, including abstracts, in line with WIPO standard ST.36. For more details see Exchange document (section 2.2.2 below) .
EPO Fulltext schema	ftxt	Describes structures for providing description, claims and drawings from a patent publication.
Cooperative Patent Classification system (CPC)	cpc	Describes structures for providing CPC. For more details see Classification service (section below) .
Register	reg	ST.36 compliant Register DTD, see: https://register.epo.org/regviewer

Table 8

2.2.2. Common response structures (XML)

Response references

Throughout the responses you will find the same structure when patent references are used (application, publication or priority):

Application-reference response:

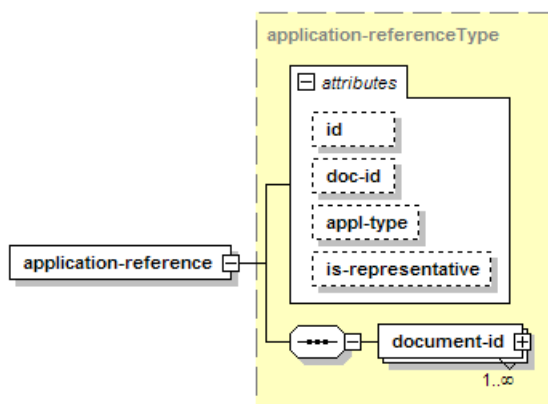


Figure 3

Application reference attributes (used in OPS responses)

Attribute name	Description
----------------	-------------

doc-id	Unique patent identifier. Please refer to the docdb user documentation for more information.
is-representative	Identifies whether this application is the family representative in the EPO simple patent family, value = YES/NO

Table 9

Publication-reference response:

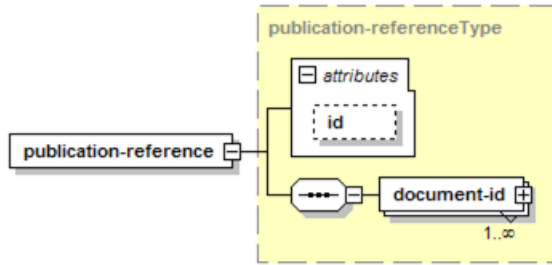


Figure 4

Priority-claim response:

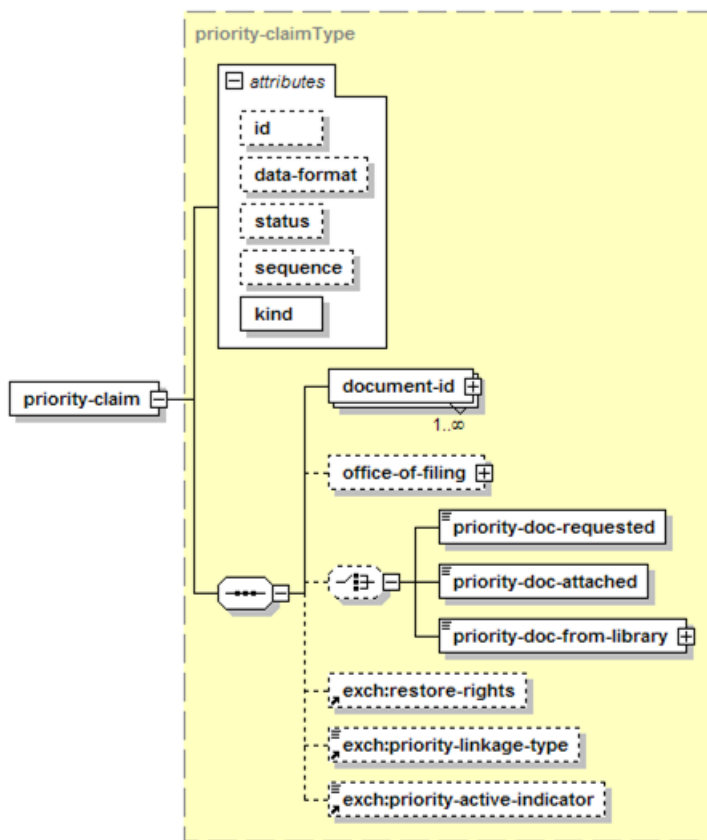


Figure 5

Priority Claims elements (optional but used in OPS family responses)

Attribute name	Description
priority-linkage-type	The Linkage Type, used to indicate the priority relation e.g. for continuation, continuation in part, for division etc.. Please refer to the docdb user documentation ANNEX III for more information.
priority-active-indicator	The DOCDB Active indicator, used to define Simple patent families. Please refer to the docdb user documentation for more information.

Table 10

Document-id

As you can see, all of the reference types use the `document-id` element, which contains the information about the reference: country code, number, kind code, date and `document-id-type` (or format):

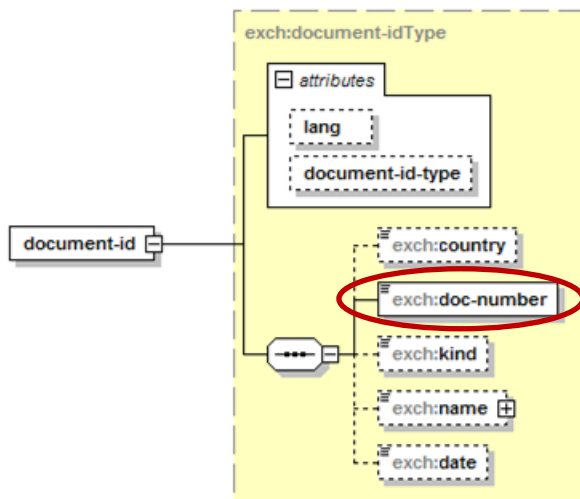


Figure 6

Note, in the schema above:

- The `doc-number` is **mandatory**.
- One reference may have **multiple** `document-id` elements in order to express the same reference in different number formats:

```
<publication-reference>
  <document-id document-id-type="docdb">
    <country>EP</country>
    <doc-number>1000000</doc-number>
    <kind>A1</kind>
    <date>20000517</date>
  </document-id>
  <document-id document-id-type="epodoc">
    <doc-number>EP1000000</doc-number>
  </document-id>
</publication-reference>
```

```

<date>20000517</date>
</document-id>
</publication-reference>

```

Exchange Document

OPS uses the [exchange-document schema](#) for the exchange of bibliographic data. For further details please see the [docdb user documentation](#).

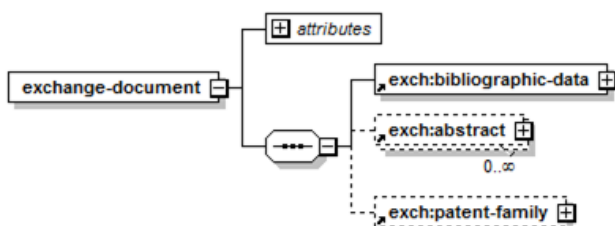


Figure 7

As shown in Figure 7 the `exchange-document` schema includes both the `bibliographic-data` and `abstract` elements. The `exch:patent-family` element is not used by OPS.

The following tables provide some highlights of the **exchange document diagram** (as shown above).

Exchange document attributes (contained in the bibliographic data response):

Attribute name	Description
system	OPS server identifier; OPS is identified by <code>ops.epo.org</code>
family-id	Unique identifier of the simple family to which the document belongs
country	Country code of the underlying document
doc-number	Document number (may have leading zeros in some cases)
kind	The document's kind code
status	Only appears when document could not be found.

Table 11

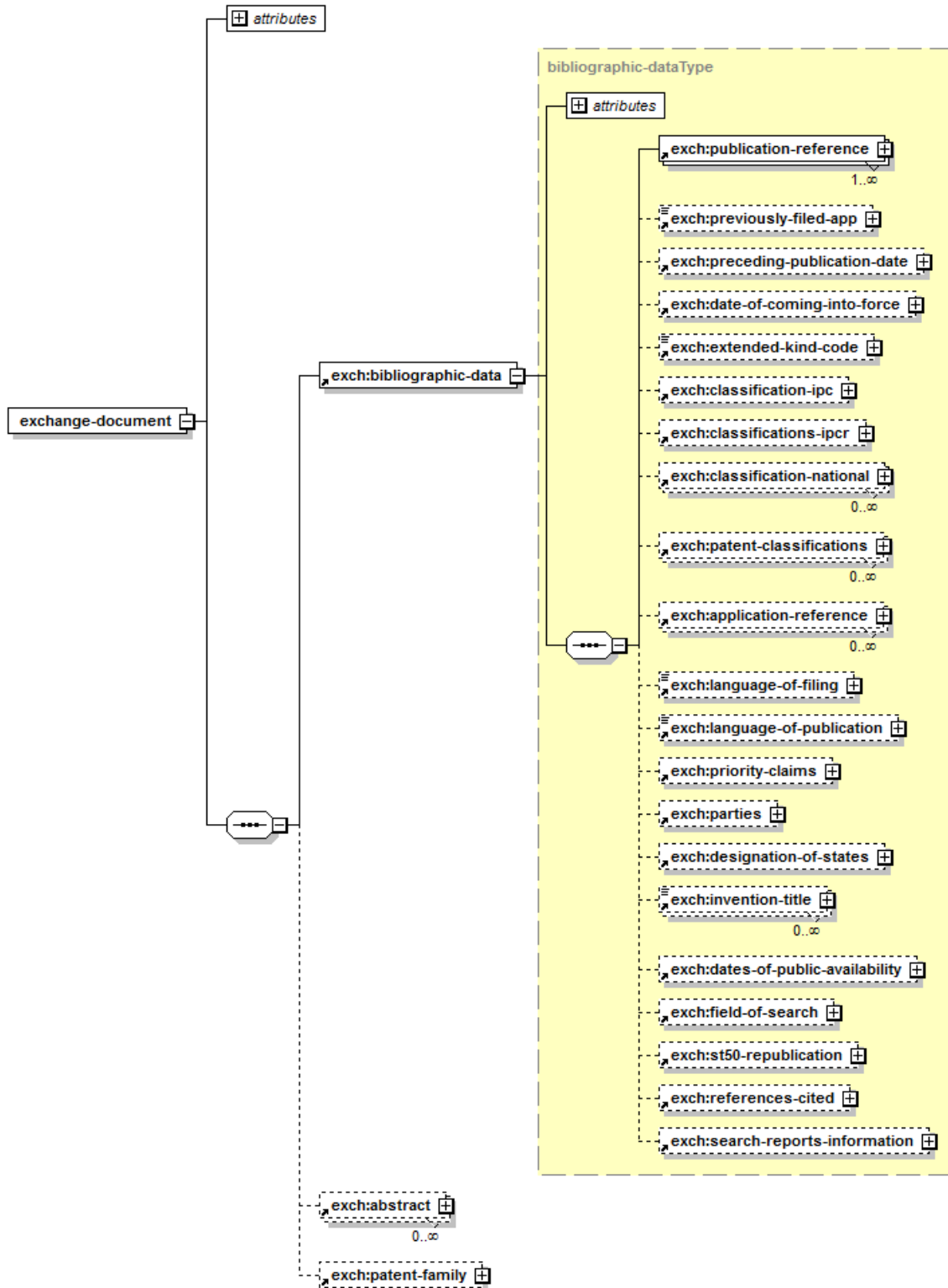


Figure 8

Some elements contained in the bibliographic-data element:

Element	Description
publication-reference	Refers to the publication in every available format. To express different formats, <code>document-id</code> elements are used (see section 'Response References'.) All fields are properly resolved. If a wildcard (*) has been used, it shall be now properly determined revealing the full kind code. Additionally, the publication date is provided in date element in the format <code>yyyyMMdd</code> , where <code>yyyy</code> refers to the publication year, <code>MM</code> to the publication month number (with the leading zero) and <code>dd</code> to the publication day number (with the leading zero).
classification-ipc	Provides the list of IPC1-7 classes the underlying publication has been classified into. <i>The content of this element is no longer supported for publication published after January 2006, please refer to classification-ipcr element instead which contains classes from IPC8</i>
classification-ipcr	Provides the list of IPC8 classes the underlying publication has been classified into, according to WIPO ST8.
patent-classification	Generic element for classifications allocated by EPO, USPTO, JPO and national offices; e.g. CPC, FI, FTERM and DOCUS. ECLA is no longer supported by the EPO and will not appear in future OPS responses.
application-reference	Refers to the application document in every available format with the application date. In most cases, <code>docdb</code> and <code>epodoc</code> format are provided, original if available.
priority-claim	Contains a priority claim document reference in the available format(s) with the priority date. The <code>sequence</code> attribute refers to the order in which priorities are printed on the published document.
applicant (epodoc format)	Contains the applicant name(s) written in <code>epodoc</code> format.
applicant (original format)	Contains the applicant name(s) as it is written on the original paper document, if available.

Element	Description
<code>inventor</code> (epodoc format)	Contains the inventor name(s) written in <code>epodoc</code> format.
<code>inventor</code> (original format)	Contains the inventor name(s) as it is written on the original paper document, if available.
<code>invention-title</code>	<p>Contains the invention title in the original language and French, German or English, if available. The language code is provided within the <code>lang</code> attribute.</p> <p>There might be many <code>invention-title</code> elements which reflect the title structure, as printed on the publication document.</p>
<code>citation</code>	<p>Provides the publication references (in <code>epodoc</code> format) to the documents considered to be relevant during a single phase of a search/grant procedure.</p> <p>The <code>sequence</code> attribute refers to the order in which documents are printed on the search report document. For more information on see citations table below.</p>
<code>abstract</code>	Contains the abstract paragraph, if available. The language code is provided within the <code>lang</code> attribute of the <code>abstract</code> element. There might be many <code>abstract</code> elements that reflect the abstract structure, as printed on the publication document.

Table 12

Citations

Citations are part of the bibliographic response ([see Table 1 in section 1.2 Patent information relevant to OPS](#)).

Structure of the citation part of the exchange document schema:

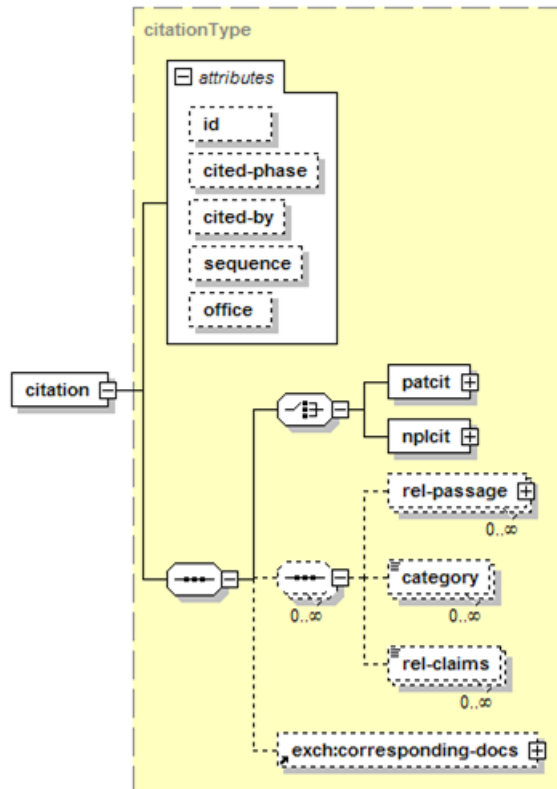


Figure 9

Note, not all of the elements that are present in the schema are actually used in the OPS response.

Citation types

There are two types of citations: patent (**patcit**) and non-patent (**npicit**).

patcit:

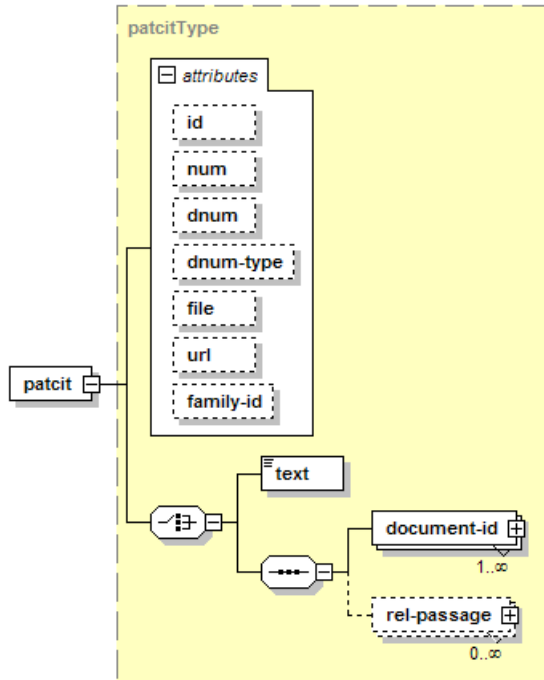


Figure 10

nplcit:

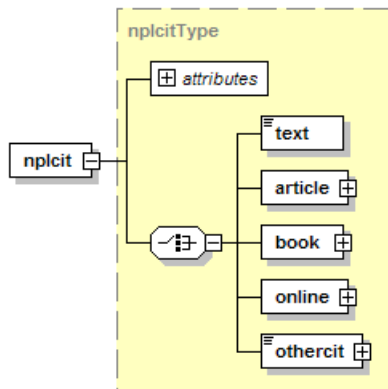


Figure 11

Citation categories

Citation categories illustrate the citation importance and are provided within the `category` element as a string combined using the following values (table below):

Letter	Description
X	particularly relevant if taken alone (- prejudicing novelty, from April 2011)
I	particularly relevant if taken alone – prejudicing inventive step (from April 2011)

Y	particularly relevant if combined with another document of the same category
A	technological background
O	non-written disclosure
P	intermediate document
T	theory or principle underlying the invention
E	earlier patent document, but published on, or after the filing date
D	document cited in the application
L	document cited for other reasons

Table 13

Originator of the citation

The attribute `cited-by` contains information on who cited the document during the `cited-phase`. The following values are possible:

- applicant
- examiner
- opponent
- third-party
- other
- unknown

Citation phases

The phase during which the citation was revealed can be found in the `cited-phase` attribute of the `citation` element. Table 14 shows all possible citation phases.

Phase name	Description
description	Cited during the description (application) phase
search	General Search phase; used when no further information available. Otherwise, use one of the concrete phases below:
international-search-report	Originates from International Search Report
supplementary-international-search-report	Originates from the international Supplementary Search Report
national-search-report	Originates from the national Search Report
supplementary-national-search-report	Originates from the national Supplementary Search Report
international-type-search-report	Originates from the international Type Search Report
examination	General Examination phase; use EXA when there is no further information available or one of the below if there is:

Phase name	Description
international-examination	Cited during PCT chapter II (preliminary examination); international examination
national-examination	Revealed during the national examination phase
opposition	Revealed during the Opposition phase
limitation	Revealed during the Limitation phase
other	Other documents of interest (cited in compte rendu but not in search report)
unknown	No information available

Table 14

2.2.3. Error messages

There are some situations when the OPS may respond with an error message. OPS error messages are always returned in XML format.

HTTP code	Error code	Error occurrence
404	CLIENT.InvalidReference	At least one reference in the request is invalid: <DETAILS>.
404	CLIENT.WrongReferenceFormatting	At least one reference in the request has a wrong value for data-format
413	CLIENT.AmbiguousRequest	The requested input was ambiguous ¹ .
404	SERVER.EntityNotFound	The requested data could not be found. <DETAILS>
404	CLIENT.NotFound	Resource not found
400	CLIENT.InvalidQuery	The request was invalid.
400	CLIENT.CQL	The query provided is invalid, <DETAILS>
405	CLIENT.MethodNotAllowed	Can't serve media types listed in the Accept header, <DETAILS>.
403	CLIENT.RobotDetected	Recent behaviour implies you are a robot. The server is at the moment busy to serve robots. Please try again later.

¹ **CLIENT.AmbiguousRequest** may happen only when requesting the information in DOCDB format. For example the request <http://ops.epo.org/rest-services/published-data/publication/epodoc/EP.0536425/fulltext> will always return the claims/description availability for the given number while <http://ops.epo.org/rest-services/published-data/publication/docdb/EP.0536425/fulltext> will result ambiguity, as given number matches several publications. The client needs to select some number and repeat the request using that specific number.

503	SERVER.LimitedServerResources	Please request bibliographic data in smaller chunks ² .
500	SERVER.DomainAccess	The request could not be processed. Please try again later.
408	CLIENT.RequestTimeout	The query resulted in a timeout on our back end. Please reformulate it to be less complex or access a smaller result set.

Table 15

2.3. Registration & OPS Fair use policy

OPS receives traffic from many hundreds of concurrent user hosts per day. The OPS service is connected to several EPO backend databases used by the EPO examining divisions. It is designed to be responsive to our external users without overwhelming the servers, thus enabling the EPO to continue its core operations unimpeded by OPS.

The Fair Use policy is required to ensure that users are aware that there are limits to the amount of data we can serve both in volume and the overall number of requests. To protect the EPO systems from malicious denial of service attacks or naive application development by 3rd parties, a number of measures are in place to encourage fair use:

Users are categorised in the following way:

- Anonymous users have no access to OPS.
- Registered users are granted free access to higher volumes of data up to a certain ceiling defined in the OPS terms and conditions.
- Registered users willing to pay for more substantial volumes of data can record Billing contact details and apply for access to higher volumes of data.

The published fair use policy and associated terms and conditions define what we consider to be appropriate usage by registered users. It is enforced in the following ways:

- Registered users must authenticate when accessing OPS using https and OAuth (described below).
- Dynamic fair use monitoring inside the OPS system gives the user feedback on usage in such a way that enables the user to control the behaviour of their client application. This feedback takes the form of HTTP headers with each response generated by OPS specifically for the user.
- If the user behaviour exceeds the terms of the fair use policy then the dynamic fair use controls ensure that access is reduced accordingly.

² **SERVER.LimitedServerResources** error might occur if user requests too many data in one go. In order to gracefully correct such error user have to reduce the size of the range if request support ranges or request one bibliographic data at a time.

- Users that are found to be especially disruptive can be blocked completely from the application until agreement is reached with OPS support to release them.

2.3.1. User Registration for access

Overview

By registering, the user will receive access credentials to the OPS service. The access credentials are used to authenticate the user and obtain an access token. With the access token, the user can access the OPS server.

Each registered developer can define one or more sets of Client Credentials and secret. This is for testing purposes to allow different versions of OPS to be used by the same developer. The terms and conditions of use relate to a single developer registration.

HTTPS is used for all communication with the registration portal and OPS when the exchange of access credentials is involved.

Prior to attempting registration the user should have the following information at hand:

- A username (must be unique to the OPS system)
- An email address (must be unique to the OPS system). This email address will be used for all registration and activation correspondence.
- An address. We require detailed information concerning your location and organisation.
- Contact details. We require telephone and email addresses where we can contact a technical representative.
- Billing details. Users requiring access to large volumes of data can select the option to pay (please refer to terms and conditions). Information required includes VAT and billing address details. Optionally a Purchase order number can be specified. Separate telephone and email contact details are required to deal with invoicing and settlement issues.

Please note: The registration details entered on this screen are unique to OPS developer registrations. The username and password recorded here should not be confused with username and passwords for the EPO Forum, the European Patent Register or indeed for any other EPO secure registration and connection.

It is the users responsibility to NOT reuse passwords that have also be used in other registered profiles in other system. For example, you should not reuse twitter, facebook, gmail, outlook, itunes or indeed other EPO passwords in this registration.

Submission of registration request

- The registration page can be found at <https://developers.epo.org>
- The user enters the registration details including Billing contact information if large volumes of data are required

- The user accepts the terms and conditions and confirms that he/she is authorised to act on behalf of the organisation being registered.
- If the user is willing to pay for substantial volumes of data, then the system will ask for confirmation that the user understands that this will result in invoices being produced.
- The user submits the registration request.
- Emails are sent to the EPO administrator and the user requesting registration.
- The account is NOT activated until the EPO approves the registration request.

Approval of registration request

- The EPO applies a number of criteria before activating the account
- As a general rule, we avoid creating duplicate or vaguely defined user accounts
- We will attempt to contact you if your details are not complete
- Emails are sent to the user when the account is approved or rejected

Creation of Client credentials

- Once the user has received notification that the registration has been approved, with a copy of the Terms and Conditions that the user accepted, the OPS Client credentials can be generated. The email includes a link to further edit the registered user profile.
- The user should select the option to define Applications (“My Apps”).
- Each App added will produce a new set of client credentials
- Each App should be given a name and be specific to one of the available OPS systems (e.g. OPS v3.1 or future versions).
- The consumer Key and Secret will appear on the App definition screen:

Consumer Key: y3AOG86qwqjU0QU69VYGTJ4JGxUN8EVG

Consumer Secret Key: rrXdr5WA7x9tudmP

Figure 12: Example Client credentials made up of Consumer Key and Consumer Secret

2.3.2. Authentication & Access Token handling

OAuth Authentication and Access tokens

OPS uses the OAuth framework for Authentication and Authorization. At this point in time, only the “Client Credentials” flow is supported using a Consumer key and Consumer secret.

The following sequence diagram shows the steps that must be followed to convert client credentials to an access token, in order to consume resource from the OPS service.

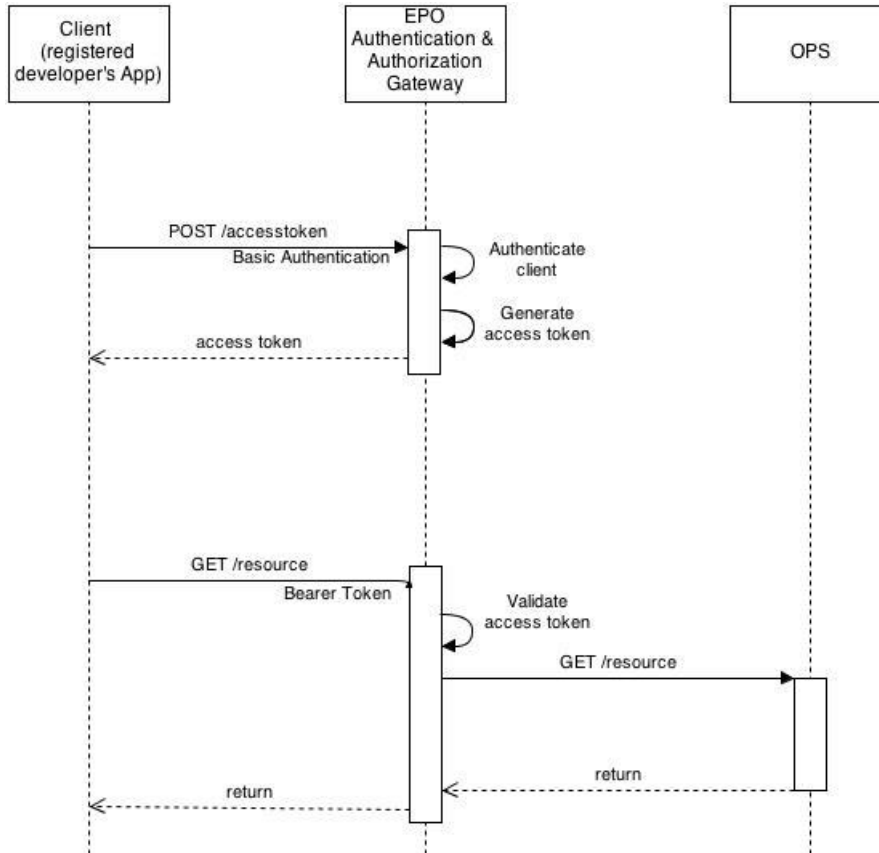


Figure 13: OAuth Access Token and OPS Request Sequence diagram

The actual steps to follow are:

Step 1: Client converts Consumer key and Consumer secret to `Base64Encode(Consumer key:Consumer secret)`.

This should be done programmatically using the language you are developing the client application in. For the purposes of this example, a public website was used to perform this conversion.

By entering the **colon separated** Client credentials, an encoded response is generated. This response is then be used for basic Authentication.

The `Form.SizeLimit` is 10000000bytes. Please, do not post more data using this form.

Base64 representation of the source data:

```
eTNBT0c4NnF3cWpVMFFVNjIWWUdU5jRKR3hVTjhFVkc6cnJYZHI1V0E3eDI0dWRtUA==
```

Type (or copy-paste) some text to a textbox bellow. The text can be Base64 string to decode or any string to encode to a Base64.

```
y3AOG86awajU0QU69VYGTJ4JGxUN8EVG:rrXdr5WA7x9tudmP
```

Figure 14: Converting colon separated Consumer Key and Consumer Secret to Base64 Encoding

Step 2: Client requests an access token using Basic Authentication, supplying its Consumer key and Consumer secret with base64Encoding over encrypted HTTPS connection:

HTTPS	POST https://ops.epo.org/3.2/auth/accesstoken
HTTP Headers	
Authorization:	Basic eTNBT0c4NnF3cWpVMFFVNj1WWUdUSjRKR3hVTjhFVkc6cnJYZHI1V0E3eD10dWRtUA==
Content-Type:	application/x-www-form-urlencoded
POST request payload	
grant_type=client_credentials	

OPS authenticates the client credentials passed in the Authorization header using basic authentication method.

If credentials are valid, OPS responds with a valid access token

<pre> HTTP/1.1 200 OK Content-Type: application/json Cache-Control: no-store Pragma: no-cache { "issued_at": "1364247843353", "application_name": "511d82a3-aa0e-4775-ba48-05ccd9275c56", "scope": "", "status": "approved", "expires_in": "1199", "api_product_list": "[ops31-prod]", "organization_id": "0", "access_token": "4AWoepfVNgf09DRmimEnGdXcgoFU", "organization_name": "epo", "refresh_count": "0" } </pre>

The access token produced is valid for approximately 20mins.

A New access token should be requested as soon as an invalid access token message is received from OPS.

Step 3: Client accesses OPS resources with access token in authorization header (bearer tokens) over encrypted HTTPS connection

HTTPS	GET https://ops.epo.org/rest-services/published-data/publication/epodoc/EP1000000
HTTP Headers	
Authorization:	Bearer 4AWoepfVNgf09DRmimEnGdXcgoFU

Error conditions

Spike/Quota Errors

When

Returned when spike or quota limit has been reached.

XML Response

```
403 - FORBIDDEN
<error>
  <code>403</code>
  <message>This request has been rejected due to the violation of Fair Use
policy</message>
  <moreInfo>http://www.epo.org/searching/free/espacenet/fair-
use.html</moreInfo>
</error>
```

Blacklisting

When

Request is made to any resource that will be blocked (blacklisting)

XML Response

```
403 - FORBIDDEN
<error>
  <code>403</code>
  <message>This request has been rejected</message>
</error>
```

Developer Blocked

When

Developer account is blocked by EPO admin.

XML Response

```
403 - FORBIDDEN
<error>
  <code>403</code>
  <message>Developer account is blocked</message>
</error>
```

OAuth Errors

When

An authentication request is made but it is failing.

XML Response

```
HTTP/1.1 400 Bad Request
Content-Type: application/xml;charset=UTF-8
Cache-Control: no-store
Pragma: no-cache
```

```
<error>
  <code>400</code>
  <message>invalid_client</message>
  <description>Developer account is blocked</description>
</error>
```

The message field entry for an authentication error can be one of the following:

invalid_request

The request is missing a required parameter, includes an unsupported parameter value (other than grant type), repeats a parameter, includes multiple credentials, utilizes more than one mechanism for authenticating the client, or is otherwise malformed.

invalid_client

Client authentication failed (e.g., unknown client, no client authentication included, or unsupported authentication method). The authorization server MAY return an HTTP 401 (Unauthorized) status code to indicate which HTTP authentication schemes are supported. If the client attempted to authenticate via the "Authorization" request header field, the authorization server MUST respond with an HTTP 401 (Unauthorized) status code and include the "WWW-Authenticate" response header field matching the authentication scheme used by the client.

unsupported_grant_type

The authorization grant type is not supported by the authorization server.

invalid_access_token

Access token provided is either invalid or expired.

"Client identifier is required" (code 401)

Basic authentication with base64Encoding is improperly formed.

“Required param : grant_type”

Content type or grant_type payload improperly formed

The Developer’s Area

A developer’s area is available at <https://developers.epo.org>. Log in and go to <https://developers.epo.org/ops-v3-2/apis> to experiment with the above concepts.

EPO OPS v3.2 API Documentation

Published




METHOD	DESCRIPTION
GET 	Published Images Retrieval Service Images retrieval service http://ops-i.epo.org/3.2/rest-services/published-data/images/{image-country}/{image-number}/{image-kind}/{image-type}
GET 	Published Data Keywords Search without Constituents Bibliographic data search service http://ops-i.epo.org/3.2/rest-services/published-data/search
GET 	Published Data Keywords Search with Variable Constituents Bibliographic data search service http://ops-i.epo.org/3.2/rest-services/published-data/{constituent}/search

Figure 15 The API Documentation

The developer’s area allows users to select OPS services and observe the exchange of request & responses. Only registered users are able to use the developer’s area.

Example steps are:

- Select a service API that you want to use: e.g. GET /published-data/search
- The Authentication model you have to use is OAuth 2.0.
NOTE: for sending requests to OPS via developer’s area it is necessary to add an app in menu option “My Apps” --> “Add a new App”. Then return back to the selected service API, set the OAuth 2.0 like it is shown in figure 16 and click “Generate OAuth Token”. After generating the token you can send requests to OPS via developer’s area.
- The developer’s area pre-populates example Query, Template and Header entries for a valid query.
- Fill all the required parameters if any is on the form.
- Click “Send This Request” to view the HTTP header request, and the HTTP header response.

On a website <https://developers.epo.org> we also offer Quick help documentation which can help with testing OPS using Developer portal tools.

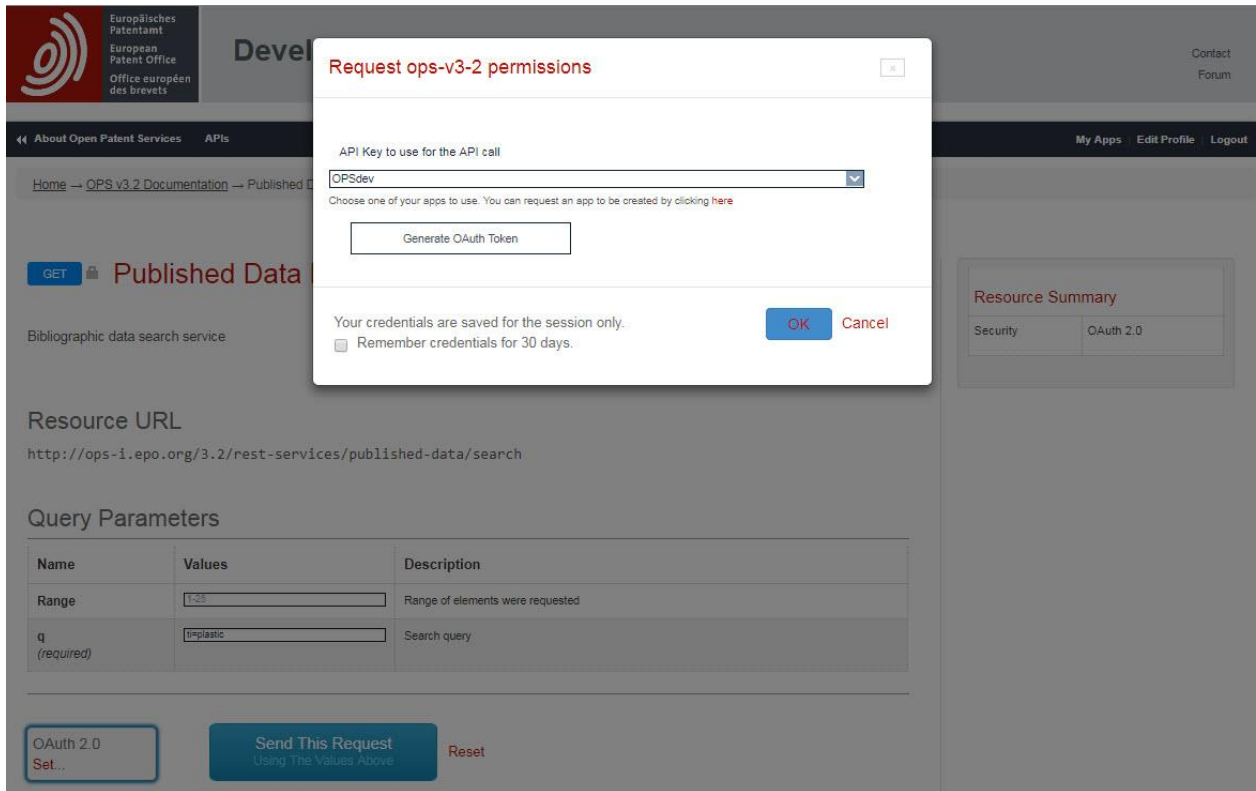


Figure 16 Request OPS permissions

2.3.3. Dynamic fair use monitoring

This chapter describes how the dynamic fair use control is applied, its effect when triggered, and what users can do to avoid it.

HTTP Headers

The OPS system makes extensive use of HTTP headers to communicate information back to the client application.

The section below describes the following HTTP headers:

- Quota Headers
 - X-IndividualQuotaPerHour-Used
 - X-RegisteredQuotaPerWeek-Used
 - X-RegisteredPayingQuotaPerWeek-Used
 - X-Rejection-Reason
- Self-throttling headers
 - X-Throttling-Control

Quota Headers

The fair use policy describes data and bandwidth thresholds. These can be summarised as:

- A global 1Mbps (megabit per second) rule for all users. This is enforced as an hourly quota equal to approx 450MB per hour.
- OPS is free of charge up to the free threshold currently set in the fair use charter described here: <https://www.epo.org/service-support/ordering/fair-use.html>.
- All excess usage above this will be Paid for usage, requiring special approval in the registration process.

The Quota related custom http headers, X-IndividualQuotaPerHour-Used, X-RegisteredQuotaPerWeek-Used, X-RegisteredPayingQuotaPerWeek-Used, are produced with relevant requests to inform the users the current state of quota consumed.

Once a specific quota is exhausted, the system will respond with a X-Rejection-Reason header.

As a general rule, Hourly quotas will be fully refreshed within 1 hour on a rolling window basis. This is to encourage users to maintain a steady usage profile. As the rolling window advances, quota consumed at the beginning of the hour is released and made available. Consequently users who develop software that sustains a very intensive request profile will find the flow of data interrupted.

Weekly quota will be released each calendar week at midnight UTC/GMT.

Example Quota header response with quota remaining

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-IndividualQuotaPerHour-Used: 3006
X-RegisteredQuotaPerWeek-Used: 900006
X-Throttling-Control: idle (retrieval=green:200, search=yellow:20,
inpadoc=red:30, images=green:200, other=green:1000)
Content-Type: application/xml
Transfer-Encoding: chunked
Date: Fri, 02 Sep 2011 06:52:11 GMT
```

Example Quota header response with quota exhausted

```
HTTP/1.1 403 Forbidden
Content-Type: application/xml
X-Rejection-Reason: RegisteredQuotaPerWeek
Content-Length: 207

<error>
  <code>403</code>
  <message>This request has been rejected due to the violation of Fair
Usepolicy</message>
  <moreInfo>http://www.epo.org/searching/free/espacenet/fair-
use.html</moreInfo>
</error>
```

Self-throttling headers

OPS tracks concurrent usage of the service and establishes that the service is in one of 3 states:

- Idle – Low concurrent usage being applied to the service at present
- Busy – a significant number of concurrent requests are being handled
- Overloaded – a very significant number of concurrent requests are being handled

For each system state, OPS applies limits to the number of requests that a single user system can make over a period of time (a 60 second window). As the usage from a single user host increases, the system is able to detect when a defined limit has been reached for that user.

This header is used to further encourage users to moderate their usage profile such that requests are sent to OPS in a uniform and consistent flow, avoiding periods of intense usage.

Example Throttling Control header response

Throttling information is provided in the http header response, indicating the system state, the traffic light status for a particular service, and information concerning the fair use limitation in place.

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-Powered-By: Servlet 2.4; JBoss-4.3.0.GA_CP07 (build:
SVNTag=JBPAPP_4_3_0_GA_CP07 date=200911251949)/JBossWeb-2.0
X-Throttling-Control: idle (retrieval=green:200, search=yellow:20,
inpadoc=red:30, images=green:200, other=green:1000)
Content-Type: application/xml
Transfer-Encoding: chunked
Date: Fri, 02 Sep 2011 06:52:11 GMT
```

The X-Throttling-Control header is specific to each user system and is delivered with every OPS response. It indicates the current system state at that point in time.

Generic throttling information structure:

```
X-Throttling-Control: system-state (service-name=traffic-light-
position:request-limit, ...)
```

The traffic light indicator for any single service can be in one of 4 positions:

- **Green** – less than 50% of the permitted request limit has been used
- **Yellow** – between 50% and 75% of the request limit has been used
- **Red** – more than 75% of the request limit has been reached
- **Black** – the limit has been exceeded and service has been temporarily suspended

The request limit is variable in 2 ways:

- Each individual service (retrieval, search, images, inpadoc or other) has different request limits

- As the system state changes from Idle to busy, or busy to overloaded, the request limit for all users is reduced.

Example:

3 successive requests could receive the following response headers:

```
X-Throttling-Control: idle (retrieval=green:200, search=green:30,
inpadoc=green:60, images=green:200, other=green:1000)
X-Throttling-Control: busy (retrieval=green:100, search=green:15,
inpadoc=green:45, images=green:100, other=green:1000)
X-Throttling-Control: overloaded (retrieval=green:50, search=green:5,
inpadoc=green:30, images=green:50, other=green:1000)
```

In between the 3 requests the system state has changed (more and more concurrent users are using the service). As a result the system has reduced the request-limit to instruct users to reduce the level of request activity.

The values are determined over a 60 second window of utilisation. In other words the traffic light and request limit reflect the number of requests a client has been making, and are entitled to make over a fixed 60 second time frame.

When the black indicator has been reached a time indicator will appear:

```
X-Throttling-Control: overloaded (retrieval=green:50, search=green:5,
inpadoc=black:0, images=green:50, other=green:1000)
Retry-After: 60000
```

Note, the Retry-After counter indicates the remaining time for which the service is suspended.

Multiple instances of OPS

The above scenario describes the response behaviour of a single instance of OPS. In fact there are many instances of the OPS service running in parallel. Each instance receives traffic from a load balancer, which distributes the requests across the system. At present the EPO has not implemented communication between OPS instances. Thus when a stream of requests are sent, a stream of responses will be received, generated by the farm of OPS server instances.

For example, if a long series of requests for Family or Legal data is made (affecting the family service):

```
http://ops.epo.org/rest-
services/family/priority/docdb/NL.1010536.A etc....
```

the responses received could resemble the following, where t represents the number of seconds passing sequentially:

```
t=0 X-Throttling-Control: idle (retrieval=green:200, search=green:30,
inpadoc=green:60, images=green:200, other=green:1000)
t=1 X-Throttling-Control: busy (retrieval=green:100, search=green:10,
inpadoc=green:15, images=green:100, other=green:1000)
```

...20 seconds later the following series of responses might be received:

```
t=20 X-Throttling-Control: overloaded (retrieval=green:50, search=green:5,
inpadoc=red:30, images=green:50, other=green:1000)
t=21 X-Throttling-Control: overloaded (retrieval=green:50, search=green:5,
inpadoc=green:30, images=green:50, other=green:1000)
t=22 X-Throttling-Control: busy (retrieval=green:100, search=green:15,
inpadoc=green:45, images=green:100, other=green:1000)
t=23 X-Throttling-Control: idle (retrieval=green:200, search=green:30,
inpadoc=green:60, images=green:200, other=green:1000)
t=24 X-Throttling-Control: idle (retrieval=green:200, search=green:30,
inpadoc=green:60, images=green:200, other=green:1000)
t=25 X-Throttling-Control: idle (retrieval=green:200, search=green:30,
inpadoc=green:60, images=green:200, other=green:1000)
```

Each request will be handled by a separate instance of OPS. From this sequence it is clear that one OPS response is reporting that a particular OPS instance is struggling with the workload. The service is overloaded and the family (inpadoc) service being called is in status **red** because this instance has served several of your requests. Although other responses received indicate idle/inpadoc:**green**, the instance that reported the problem may still be in the state reported. As a user adhering to the fair use policy, you should moderate your usage of the service over a 60 second window, according to the most negative response data received. Self-throttling correctly will enable you to avoid temporary or permanent suspension by the OPS instance.

Mapping between services and throttles

Throttle	Rest service URI	SOAP service
search	/published-data/search/*	biblio-search
retrieval	/published-data/*	biblio-retrieval
inpadoc	/family/*	family-* retrieval
inpadoc	/legal/*	legal-retrieval
images	/published-data/images/*	document-retrieval
images	/classification/cpc/media/*	

Table 16

All other OPS services not included in the list above have throttle “other”.

2.3.4. Data usage API

Overview

For users who have registered for OPS and particularly those who consume paid-for data, OPS now has an API to assist in tracking the usage of data. The response produced by this API is based on the same database used for invoicing purposes by the EPO.

The data usage report is updated within 10 mins of each hour. A request made at 10.01 will be reflected in the report between 11.00 and 11.10. The report gives a break down day-by-day and is aligned on midnight UTC/GMT.

Request structure

The user should request a valid access token prior to calling the data usage API.

A single date, or a range of dates can be passed to the service. If a data range is given, the dates should be separated with a ~ [tilde] character.

Calls to the data usage API are not subject to charging and consequently do not contribute to the results presented.

GET

[https://ops.epo.org/3.2/developers/me/stats/usage?timeRange=dd/mm/yyyy\[~dd/mm/yyyy\]](https://ops.epo.org/3.2/developers/me/stats/usage?timeRange=dd/mm/yyyy[~dd/mm/yyyy])

Authorization: **Bearer 4AWoepfVNgf09DRmimEnGdXcgoFU**

For example for 30th September to 1st October:

GET

<https://ops.epo.org/3.2/developers/me/stats/usage?timeRange=30/09/2013~1/10/2013>

Authorization: **Bearer 4AWoepfVNgf09DRmimEnGdXcgoFU**

Response structure

The response produced is in JSON format.

```
{
  "environments": [
    {
      "dimensions": [
        {
          "metrics": [
            {
              "name": "total_response_size",
              "values": [
                {
                  "timestamp": 1380585600000,
                  "value": "1.1781815E7"
                },
                {
                  "timestamp": 1380499200000,
                  "value": "58579.0"
                }
              ]
            },
            {
              "name": "message_count",
              "values": [
                {
                  "timestamp": 1380585600000,
                  "value": "420.0"
                },
                {
                  "timestamp": 1380499200000,
                  "value": "6.0"
                }
              ]
            }
          ]
        },
        {
          "name": "123456,Registered Developer,4301"
        }
      ],
      "name": "prod"
    }
  ],
  "metaData": {
    "errors": [],
    "failedEnvs": "[]",
    "notices": []
  }
}
```

total_response_size (Byte) – represents the actual and full amount of data consumed by this user on the given date. It includes the amount of free data as well as paid-for data. Paying users will receive an invoice that has been adjusted to allow for the free data quota. In cases of large amounts of data, the value is given in scientific notation:

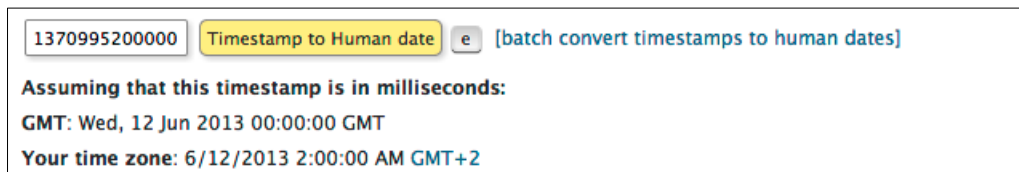
e.g. $1.1781815E7 = 1.1781815 \times 10^7$ or 11781815 Bytes.

message_count – indicates the number of requests received from this developer registration to produce the data response size indicated.

Timestamp is given with the Unix time for the dates included in the response.

The following website can be used to decode the Unix time manually:

<http://www.epochconverter.com/>



1370995200000 **Timestamp to Human date** e [batch convert timestamps to human dates]

Assuming that this timestamp is in milliseconds:
GMT: Wed, 12 Jun 2013 00:00:00 GMT
Your time zone: 6/12/2013 2:00:00 AM GMT+2

Figure 17: Unix time converted to Human readable time

Results are grouped for the developer and date range give.

The response indicates the registered user's details from the registration record:

```
"name": "123456,Registered Developer,4301"
```

These fields are the EPO account number, the Registered developer name and the unique registration record number respectively.

2.3.5. Encryption protocols

Open Patent Services version 3.2 fully supports TLS 1.2 protocol and for backward compatibility temporary supports SSL 3 and TLS 1.0/1.1.

3. OPS SERVICES

- 3.1. Published-data services
- 3.2. Family service
- 3.3. Number-service
- 3.4. Register service
- 3.5. Legal service
- 3.6. Classification services

[top of document](#)

3.1. Published-data services

In this section you will find the services for retrieving published data.

Note, some services require a two-step process: inquire, then retrieve, especially for `fulltext` and `images`.

Through different endpoints the **published-data service** provides access to worldwide patent data:

Endpoints	Description
<code>fulltext</code>	Inquires about whether claims and/or descriptions are available for patents.
<code>claims</code>	Retrieves the patent claims .
<code>description</code>	Retrieves the patent description .
<code>images</code>	Inquiry and retrieval of images or "drawings", which can also be PDF documents.
<code>equivalents</code>	Retrieves the simple patent family ; patent documents that share the same technical information, i.e. across different countries. This endpoint also supports requests that use constituents (see Table 18 below.)
<code>biblio</code>	The default response data for the published-data service, therefore the <code>biblio</code> endpoint is used for retrieval of and searches for bibliographic data . This endpoint is actually to support requests for constituents (see next table for details.)
<code>abstract</code>	Retrieves the patent abstract of the bibliographic data.

Table 17: Endpoints

Published-data constituents

For the `biblio` and `equivalents` endpoint requests, retrieval possibilities are extended with "constituents", which modify what you will get in your responses.

Constituents	Delivers
full-cycle	the full publication cycle of the patent application. This consists of a "full set" or all of the publications related to an application.
images	drawings and descriptions of the patent (.png, .pdf, .tiff)
biblio	bibliographic data
abstract	patent abstract – This is only a constituent when the <code>equivalents</code> endpoint is used in the request (see Table 17: Endpoints above.)

Table 18

Generic published-data request structure

In the request structure below, **bold means mandatory**.

```
GET http://ops.epo.org/rest-services/ published-data/ reference-
type publication or application or priority/input-format docdb or
epodoc/input e.g. EP1000000 or EP1000000.A1/[endpoint e.g.
biblio, abstract, equivalents, fulltext, claims, description, images]
and-or /[constituent e.g. biblio, full-cycle, abstract]?[parameter e.g.
Range=1]

Accept: application/exchange+xml
```

Note, a published-data request without an endpoint provides the bibliographic data by default or the data if you added the '/biblio' endpoint.

Note, in published-data requests, the POST method is also supported. Instead of giving the input in the request URL, it has to be provided in the HTTP body. You will find examples in the following chapters alongside the GET example requests.

Combinations of endpoints and constituents

In the published-data service there are specific combinations of endpoints and constituents.

Valid constituents in service requests:

Services	Endpoints	Constituents	Can be combined
Published-data services	/ ³	/biblio	+
		/abstract	+
		/full-cycle	+
	/image		
	/fulltext		
	/claims		
	/description		
	/equivalents ⁴	/biblio	+
		/abstract	+
		/full-cycle	+
Bibliographic search	/ ⁴	/biblio	+
		/abstract	+
		/full-cycle	+
Family service	/ ⁴	/biblio	+
		/legal	+
Register retrieval	/ ³	/biblio	+
		/events	+
		/procedural-steps	+
Register search	/ ⁴	/biblio	+
		/events	+
		/procedural-steps	+

Table 19

Examples of generic combinations (specific request examples are included in their respective sub-sections):

```
.../<input>/equivalents/biblio,full-cycle,abstract
.../<input>/biblio,full-cycle,abstract
```

Notes:

Constituents included in requests must be separated with commas (examples are included in the sub-sections.)

The `/images` endpoint, for images retrieval requests is followed by the link contained in the XML response of the images inquiry ([section 3.1.3](#)).

³ no constituent (/) defaults to biblio request

⁴ no constituent (/) results list of numbers

3.1.1. Bibliographic data

Retrieves the bibliographic data of all patent reference types, including bulk retrievals.

Biblio retrieval (simple, in bulk and/or with full-cycle)

Request example:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000.A1/biblio
```

```
Accept: application/exchange+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/biblio
```

```
Accept: application/exchange+xml
```

```
Request Body: EP1000000.A1
```

Note, by default a request that does not include a constituent will automatically return "biblio" data. OPS will redirect requests without a constituent to appear the request example above. Furthermore, when request `biblio`, the `abstract` is automatically included in the response, if available.

Note, kind codes in requests affect responses:

- **WITH a kind code** you get only that specific publication.
- **WITHOUT** a kind code, the service gives you all of the relevant publications to the reference type.

Response structure

The Published-data service uses the [Exchange Document structure in section 2.2.2](#) to provide bibliographic data.

Response (from the web browser):

If you want to see the XML use the "View source" option in your browser. (This is true for all "styled" responses from OPS.)

EP 1000000 (A1) – Apparatus for manufacturing green bricks for the brick manufacturing industry**Bibliographic Data**

system: ops.epo.org - family-id: 19768124 - country: EP - doc-number: 1000000 - kind: A1

Publication Reference: DOCDB:	Country:	EP	
	Doc-number:	1000000	
	Kind:	A1	
	Date:	20000517	
EPODOC:	Doc-number:	EP1000000	
	Date:	20000517	
Classifications:	IC:	B28B5/02; B28B1/29; B28B7/00	
	IPC:	B28B 1/ 29 (B28B 1/ 29 A I); B28B 5/ 02 (B28B 5/ 02 A I); B28B 7/ 00 (B28B 7/ 00 A I); H02P 6/ 08 (H02P 6/ 08 A I)	
	EC:	H02P6/08; B28B1/29; B28B5/02B2; B28B7/00F	
Application Reference: RID:		17397285	
	EPODOC: Doc-number:	EP19990203729	
	Date:	19991108	
ORIGINAL:	Doc-number:	99203729	
Priority Claims:	EPODOC:	1) Doc-number:	NL19981010536
		Date:	19981112
	ORIGINAL:	1) Doc-number:	NL19981010536
		Date:	19981112
	EPODOC:	1) Doc-number:	1010536
		ORIGINAL:	1) Doc-number:
Parties:	Applicants:	EPODOC:	1) BOER BEHEER NIJMEGEN BV DE [NL]
		ORIGINAL:	1) BEHEERMAATSCHAPPIJ DE BOER NIJMEGEN B.V
	Inventors:	EPODOC:	1) KOSMAN WILHELMUS JACOBUS MARIA [NL]
		ORIGINAL:	1) KOSMAN, WILHELMUS JACOBUS MARIA
Invention Titles	Français:	Dispositif pour la fabrication de briques crues utilisées dans l'industrie manufacturière des briques	
	Deutsch:	Vorrichtung zur Herstellung von Steinformlingen für die Ziegelindustrie	
	English:	Apparatus for manufacturing green bricks for the brick manufacturing industry	

Figure 18

Note, all OPS services return a response in the browser that is styled with XSL but you can see the XML by using the 'view source' function.

Biblio bulk retrieval

There are two ways of requesting biblio data for multiple references:

If you want to request a small number of references, you can use the GET method:

```
GET http://ops.epo.org/rest-services/published-data/publication/epodoc/EP1676595,JP2006187606,US2006142694/biblio
```

```
Accept: application/exchange+xml
```

Response:

```
<exchange-documents>
  <exchange-document system="ops.epo.org" family-id="35840242"
country="EP" doc-number="1676595" kind="A1">
```

```

    <bibliographic-data>
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>EP</country>
          <doc-number>1676595</doc-number>
          <kind>A1</kind>
          <date>20060705</date>
        </document-id>
        <document-id document-id-type="epodoc">
          <doc-number>EP1676595</doc-number>
          <date>20060705</date>
        </document-id>
      </publication-reference>
    <!-- remaining bibliographic data stripped out for readability -->
  </bibliographic-data>
</exchange-document>
<exchange-document system="ops.epo.org" family-id="35840242"
country="JP" doc-number="2006187606" kind="A">
  <bibliographic-data>
    <publication-reference>
      <document-id document-id-type="docdb">
        <country>JP</country>
        <doc-number>2006187606</doc-number>
        <kind>A</kind>
        <date>20060720</date>
      </document-id>
      <document-id document-id-type="epodoc">
        <doc-number>JP2006187606</doc-number>
        <date>20060720</date>
      </document-id>
    </publication-reference>
  <!-- remaining bibliographic data stripped out for readability -
->
    </bibliographic-data>
  </exchange-document>
<exchange-document system="ops.epo.org" family-id="35840242"
country="US" doc-number="2006142694" kind="A1">
  <bibliographic-data>
    <publication-reference>
      <document-id document-id-type="docdb">
        <country>US</country>
        <doc-number>2006142694</doc-number>
        <kind>A1</kind>
        <date>20060629</date>
      </document-id>
      <document-id document-id-type="epodoc">
        <doc-number>US2006142694</doc-number>
        <date>20060629</date>
      </document-id>
    </publication-reference>
  <!-- remaining bibliographic data stripped out for readability -
->
    </bibliographic-data>
  </exchange-document>
</exchange-documents>

```

For large numbers of input references we recommend using the **POST** method:

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/biblio
```

```
Accept: application/exchange+xml
```

```
Request Body: EP1676595
```

```
JP2006187606
```

```
US2006142694
```

The response will be the same as using the GET method.

In order to separate multiple input references in a POST request, either use comma or new line.

Note, the bulk retrieval is limited to 100 references.

Full-cycle

Full-cycle requests the full publication cycle of the bibliographic data. With or WITHOUT a kind code in the request, the service gives you all of the publications. (See 'Patent lifecycle' diagram, section 1.3.)

Request example:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000.A1/biblio,full-cycle
```

```
Accept: application/exchange+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/biblio,full-cycle
```

```
Accept: application/exchange+xml
```

```
Request Body: EP1000000.A1
```

Response:

Note, the response includes A1 and B1 publications of the full-cycle. A biblio response automatically includes the abstract.

Abstract of EP 1000000 (A1)**English**

The invention relates to an apparatus (1) for manufacturing green bricks from clay for the brick manufacturing industry, comprising a circulating conveyor (3) carrying mould containers combined to mould container parts (4), a reservoir (5) for clay arranged above the mould containers, means for carrying clay out of the reservoir (5) into the mould containers, means (9) for pressing and trimming clay in the mould containers, means (11) for supplying and placing take-off plates for the green bricks (13) and means for discharging green bricks released from the mould containers, characterized in that the apparatus further comprises means (22) for moving the mould container parts (4) filled with green bricks such that a protruding edge is formed on at least one side of the green bricks. <IMAGE>

EP 1000000 (B1) – Apparatus for manufacturing green bricks for the brick manufacturing industry

system: ops.epo.org - family-id: 19768124 - country: EP - doc-number: 1000000 - kind: B1

Bibliographic Data

Publication Reference: DOCDB:	Country:	EP
	Doc-number:	1000000
	Kind:	B1
	Date:	20030212
EPODOC:	Doc-number:	EP1000000
	Date:	20030212
Classifications:	IC:	B28B5/02; B28B1/29; B28B7/00
	IPCR:	B28B 1/ 29 (B28B 1/ 29 A I); B28B 5/ 02 (B28B 5/ 02 A I); B28B 7/ 00 (B28B 7/ 00 A I); H02P 6/ 08 (H02P 6/ 08 A I)
Application Reference: RID :		17397285
	EPODOC:	Doc-number: EP19990203729
	Date:	19991108
Priority Claims:	ORIGINAL:	Doc-number: 99203729
	EPODOC:	1) Doc-number: NL19981010536
		Date:
Parties:	Applicants:	EPODOC: 1) BEHEERMIJ DE BOER NIJMEGEN B V [NL]
		ORIGINAL: 1) BEHEERMAATSCHAPPIJ DE BOER NIJMEGEN B.V
	Inventors:	EPODOC: 1) KOSMAN WILHELMUS JACOBUS MARIA [NL]
ORIGINAL: 1) KOSMAN, WILHELMUS JACOBUS MARIA		

Figure 19

Abstract

Abstract is a sub-element of the bibliographic data retrieval. It can be requested by itself.

Request example:

```
GET http://ops.epo.org/rest-services/published-
data/publication/epodoc/EP1000000.A1/abstract
```

```
Accept: application/exchange+xml
```

```
POST http://ops.epo.org/rest-services/published-
data/publication/epodoc/abstract
```

```
Accept: application/exchange+xml
```

```
Request Body: EP1000000.A1
```

Response:

EP 1000000 (A1) –

Bibliographic Data

system: ops.epo.org - country: EP - doc-number: 1000000 - kind: A1

Publication Reference: DOCDB:

Country: EP

Doc-number: 1000000

Kind: A1

Date: 20000517

EPODOC: Doc-number: EP1000000

Date: 20000517

Classifications:

Parties:

Abstract of EP 1000000 (A1)

English

The invention relates to an apparatus (1) for manufacturing green bricks from clay for the brick manufacturing industry, comprising a circulating conveyor (3) carrying mould containers combined to mould container parts (4), a reservoir (5) for clay arranged above the mould containers, means for carrying clay out of the reservoir (5) into the mould containers, means (9) for pressing and trimming clay in the mould containers, means (11) for supplying and placing take-off plates for the green bricks (13) and means for discharging green bricks released from the mould containers, characterized in that the apparatus further comprises means (22) for moving the mould container parts (4) filled with green bricks such that a protruding edge is formed on at least one side of the green bricks. <IMAGE>

Figure 20

Bibliographic search

The bibliographic search uses **common query language (CQL)** to retrieve the data. Additional query details are in the in the [Appendix 4.2 CQL index catalogue](#).

Query example: "applicant=IBM"

A very basic query will return a list of publication references:

```
GET http://ops.epo.org/rest-services/published-data/search?q=applicant%3DIBM
```

```
Accept: application/exchange+xml
```

```
POST http://ops.epo.org/rest-services/published-data/search
```

```
Accept: application/exchange+xml
```

```
Request Body: q=applicant%3DIBM
```

Additionally, the request can be combined with the possible constituents: abstract, biblio and/or full-cycle:

```
GET http://ops.epo.org/rest-services/published-data/search/abstract,biblio,full-cycle?q=applicant%3DIBM
```

```
Accept: application/exchange+xml
```

```
POST http://ops.epo.org/rest-services/published-data/search/abstract,biblio,full-cycle
```

```
Accept: application/exchange+xml
```

```
Request Body: q=applicant%3DIBM
```

Note, the "invalid result" status is given when the specific result found in the internal EPO system cannot be provided to external users (due to copyright or other reasons).

Response structure:

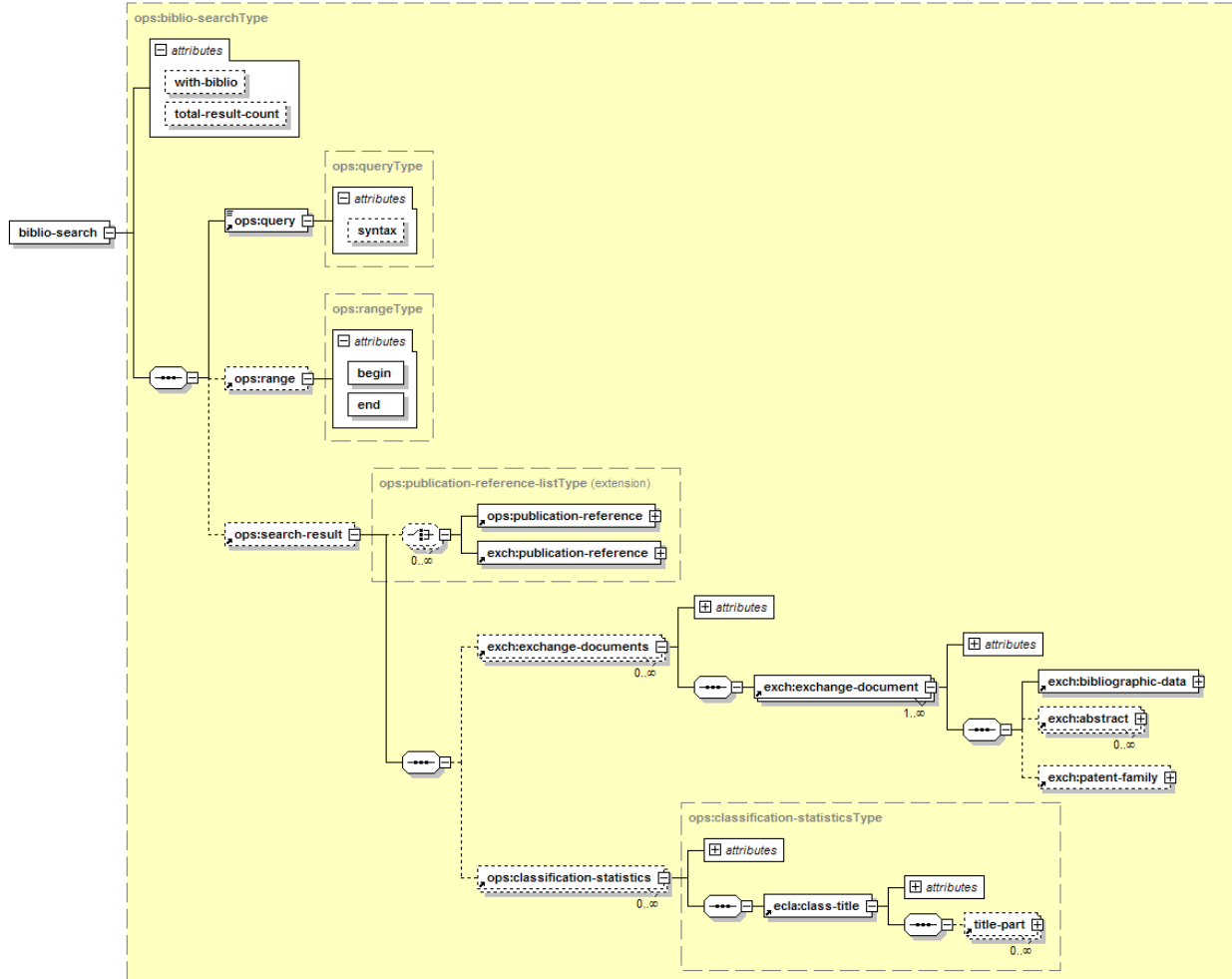


Figure 21

The structure shown in Figure 21 is part of the `ops` namespace and defined in the [ops.xsd](#) schema file.

Number of returned results

The number of matched entries in the database is returned in attribute `total-result-count`. However if this number exceeds 10000, is it truncated to this value and `total-result-count` will have value 10000. The matched database entries which cannot be retrieved due to security reasons are replaced with “This record is not available through OPS” placeholders. This is done to correctly support the retrieval of any subrange within the range `[1..total-result-count]`.

Response example (with biblio):

```

<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:biblio-search total-result-count="10000">
    <ops:query syntax="CQL">applicant=IBM</ops:query>
    <ops:range begin="1" end="25"/>
    <ops:search-result>
      <exchange-documents>
        <exchange-document system="ops.epo.org" family-id="37717388"
country="KR" doc-number="20100130646" kind="A">
          <bibliographic-data>
            <publication-reference>
              <document-id document-id-type="docdb">
                <country>KR</country>
                <doc-number>20100130646</doc-number>
                <kind>A</kind>
                <date>20101213</date>
              </document-id>
              <document-id document-id-type="epodoc">
                <doc-number>KR20100130646</doc-number>
                <date>20101213</date>
              </document-id>
            </publication-reference>
            <!-- classifications removed -->
            <application-reference doc-id="331447793">
              <document-id document-id-type="epodoc">
                <doc-number>KR20107026618</doc-number>
                <date>20060720</date>
              </document-id>
              <document-id document-id-type="original">
                <doc-number>1020107026618</doc-number>
              </document-id>
            </application-reference>
            <!-- claims removed -->
            <parties>
              <applicants>
                <applicant sequence="1" data-format="epodoc">
                  <applicant-name>
                    <name>IBM [US]</name>
                  </applicant-name>
                </applicant>
                <applicant sequence="1" data-format="original">
                  <applicant-name>
                    <name>INTERNATIONAL BUSINESS MACHINES CORPORATION</name>
                  </applicant-name>
                </applicant>
              </applicants>
              <!-- inventors removed -->
            </parties>
            <invention-title lang="en">INJECTION MOLDED MICROLENSES FOR
OPTICAL INTERCONNECTS</invention-title>
          </bibliographic-data>
        </exchange-document>
      </exchange-documents>

```

```
<!-- other search results (exchange-documents) removed for readability -->
</ops:search-result>
</ops:biblio-search>
</ops:world-patent-data>
```

Range control

The default range of the biblio search response is 1-25. It can be changed by specifying an HTTP header parameter **X-OPS-Range=[begin_range-end_range]**.

For test purposes, it is also possible to use the range query parameter by adding **&Range=[begin_range-end_range]** to the end of your URI (e.g. <http://ops.epo.org/rest-services/published-data/search?q=applicant%3DIBM&Range=50-60>).

Request Example

```
GET http://ops.epo.org/rest-services/published-data/search?q=applicant%3DIBM
Accept: application/exchange+xml
X-OPS-Range: 50-60
```

Response:

```
<ops:biblio-search total-result-count="10000">
  <ops:query syntax="CQL">applicant=IBM</ops:query>
  <ops:range begin="50" end="60"/>
```

Note, the maximum range is 100. The maximum total number of results is 2000. If there are more than 2 000 hits in your results list, you will have to limit your search query with more precise search terms to reduce the number of results.

3.1.2.Fulltext inquiry and retrieval including description or claims

Fulltext inquiry

This service provides information on what kind of fulltext is available for the given input.

Note, Currently full texts (description and/or claims) are available for the following authorities: EP, WO, AT, BE, BG, CA, CH, CY, CZ, DK, EE, ES, FR, GB, GR, HR, IE, IT, LT, LU, MC, MD, ME, NO, PL, PT, RO, RS, SE and SK.

Request example:

```
GET http://ops.epo.org/rest-services/published-data/publication/epodoc/EP1000000/fulltext
```

Accept: application/fulltext+xml

POST <http://ops.epo.org/rest-services/published-data/publication/epodoc/fulltext>

Accept: application/fulltext+xml

Request Body: EP1000000

Response structure:

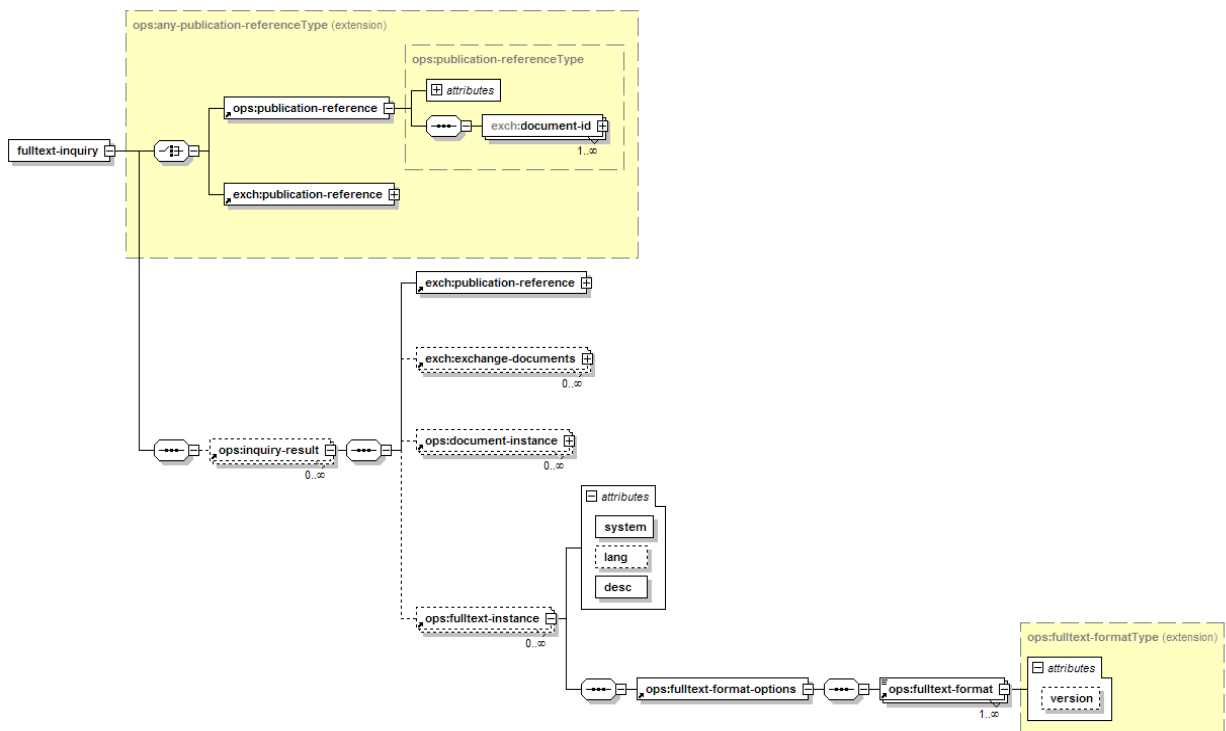


Figure 22

The structure shown in Figure 22 is part of the `ftxt` namespace and defined in the [fulltext-documents.xsd](#) schema file.

Response example:

You will see that the `fulltext` inquiry response tells you if the input, e.g. EP100000 has a description and/or claims that the service may also provide in further requests.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:fulltext-inquiry>
    <ops:publication-reference>
      <document-id document-id-type="epodoc">
        <doc-number>EP1000000</doc-number>
      </document-id>
    </ops:publication-reference>
    <ops:inquiry-result>
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>EP</country>
          <doc-number>1000000</doc-number>
          <kind>A1</kind>
        </document-id>
      </publication-reference>
      <ops:fulltext-instance system="ops.epo.org" lang="EN"
desc="description">
        <ops:fulltext-format-options>
          <ops:fulltext-format>text-only</ops:fulltext-format>
        </ops:fulltext-format-options>
      </ops:fulltext-instance>
      <ops:fulltext-instance system="ops.epo.org" lang="EN" desc="claims">
        <ops:fulltext-format-options>
          <ops:fulltext-format>text-only</ops:fulltext-format>
        </ops:fulltext-format-options>
      </ops:fulltext-instance>
    </ops:inquiry-result>
  </ops:fulltext-inquiry>
</ops:world-patent-data>
```

Fulltext retrieval

There are two types of fulltext: `description` and `claims`. Both are sub-elements of the `fulltext-document` element in the `ftxt` namespace, shown below.

Depending on what you request, one of the elements will be part of the XML response.

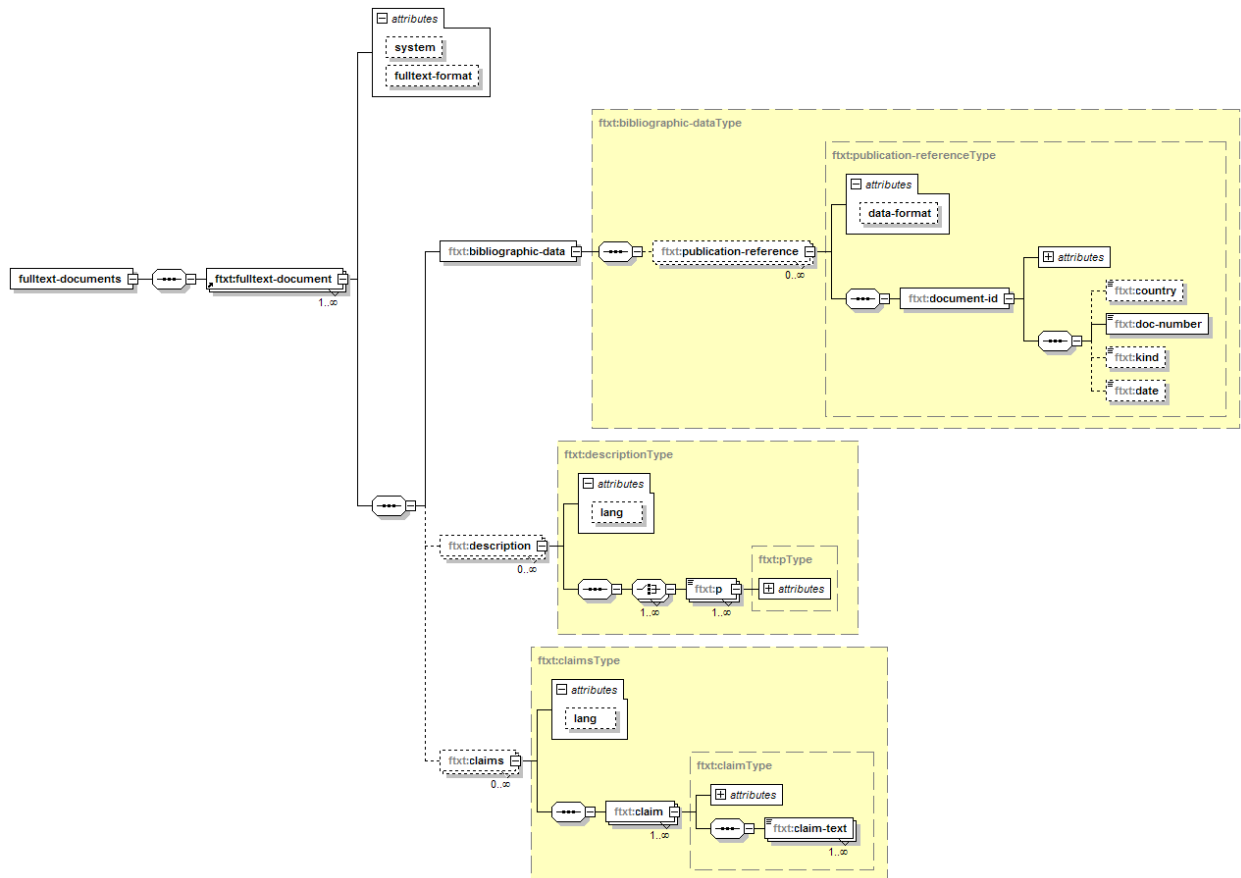


Figure 23

The structure shown in Figure 23 is part of the `ftxt` namespace and defined in the [fulltext-documents.xsd](#) schema file.

Retrieval of description

According to the fulltext inquiry, if the `description` is available, you may build this request by adding `description` at the end of the request URI.

Request example:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000/description
```

```
Accept: application/fulltext+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/description
```

```
Accept: application/fulltext+xml
```

```
Request Body: EP1000000
```

Automatically, the fulltext format will be used in the **response**:

Fulltext Document > Description

system: ops.epo.org - fulltext-format: text-only

Publication Reference: DOCDB: Country: EP

Doc-number: 1000000

Kind: A1

Description: **[@attributes]:** lang: EN

[0001] The invention relates to an apparatus for manufacturing green bricks from clay for the brick manufacturing industry, comprising a circulating conveyor carrying mould containers combined to mould container parts, a reservoir for clay arranged above the mould containers, means for carrying clay out of the reservoir into the mould containers, means for pressing and trimming clay in the mould containers, means for supplying and placing take-off plates for the green bricks and means for discharging green bricks released from the mould containers. Such an apparatus is known in the field and is for instance described in the patent 1000186 of applicant. The known apparatus is extremely suitable for automated production of large numbers of green bricks for the brick manufacturing industry. The bricks fired from these green bricks have a substantially smooth, uniform appearance.

[0002] A recent demand has developed on the market for bricks which appear as if they have been manufactured according to traditional methods.

[0003] The invention has for its object to adapt the known apparatus such that it can produce in automated manner large numbers of green bricks with a traditional appearance.

[0004] For this purpose the apparatus according to the invention has the feature that the apparatus further comprises means for moving the mould container parts filled with green bricks such that a protruding edge is formed on at least one side of the green bricks.

[0005] The bricks fired from the green bricks produced using the apparatus according to the invention impart beautiful shadow effects to the wall into which they have been built when the sun shines thereon. This aesthetic effect is an important commercial advantage.

[0006] The edge-forming means are preferably adapted to move the mould container parts repeatedly for a certain period. Repetition a number of times, for instance three times, is found in practice to be sufficient to obtain the intended effect.

[0007] In a practical preferred embodiment the edge-forming means are adapted to move the mould container parts substantially transversely of the transporting direction.

[0008] In a further preferred embodiment the edge-forming means comprise a frame which is adapted to engage individually on a mould container part. This preferred embodiment has the significant advantage that the edge-forming means can act on one mould container part while another mould container part undergoes another operation and is for instance filled with clay. The edge-forming means can therefore be added to the known apparatus without this affecting the production time.

[0009] In yet another preferred embodiment the frame spans the mould container part and is provided on both sides with stop members which are situated during operation at the location of the side walls of the mould container part. An exceptionally compact embodiment of the invention is hereby realized which utilizes the available space economically and can be arranged without difficulty on the known apparatus.

[0010] In order to prevent unnecessary damage to the mould container parts, these latter are provided on their side walls with stop surfaces, preferably of plastic. In preference the stop members of the frame of the edge-forming means are also provided with these, preferably plastic, stop surfaces.

[0011] The invention is described in more detail hereinbelow with reference to the drawing in which:

figure 1 shows schematically a preferred embodiment of the apparatus according to the invention;
 figure 2 shows in more detail a perspective view of a part of the apparatus of figure 1 with the edge-forming means therein;
 figure 3 shows the edge-forming means of figure 2 in even more detail;
 figure 4 is a perspective view of a first preferred embodiment of a mould container part which is suitable for use in the apparatus according to the invention;
 figure 5 is a perspective view of a second preferred embodiment of a mould container part; and
 figure 6 shows schematically a part of a wall which has been built using bricks provided with an edge and fired from the green bricks manufactured using the apparatus according to the invention.

Figure 24

Retrieval of claims

According to the fulltext inquiry, if the `claims` are available, you may build this request by adding `claims` at the end of the request URI.

Request example:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000/claims
```

```
Accept: application/fulltext+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/claims
```

```
Accept: application/fulltext+xml
Request Body: EP1000000
```

As in the `description` request, the `fulltext` format will be automatically used in the **response**:

[Fulltext Document > Description](#)

system: ops.epo.org - fulltext-format: text-only

Publication Reference: DOCDB:	Country:	EP
	Doc-number:	1000000
	Kind:	A1
Claims:	[attributes]: lang: EN	
)	<p>1. Apparatus for manufacturing green bricks from clay for the brick manufacturing industry, comprising a circulating conveyor carrying mould containers combined to mould container parts, a reservoir for clay arranged above the mould containers, means for carrying clay out of the reservoir into the mould containers, means for pressing and trimming clay in the mould containers, means for supplying and placing take-off plates for the green bricks and means for discharging green bricks released from the mould containers, characterized in that the apparatus further comprises means for moving the mould container parts filled with green bricks such that a protruding edge is formed on at least one side of the green bricks.</p> <p>2. Apparatus as claimed in claim 1, wherein the edge-forming means are adapted to move the mould container parts repeatedly for a certain period.</p> <p>3. Apparatus as claimed in claim 1 or 2, wherein the edge-forming means are adapted to move the mould container parts substantially transversely of the transporting direction.</p> <p>4. Apparatus as claimed in any of the foregoing claims, wherein the edge-forming means comprise a frame which is adapted to engage individually on a mould container part.</p> <p>5. Apparatus as claimed in claim 4, wherein the frame spans the mould container part and is provided on both sides with stop members which are situated during operation at the location of the side walls of the mould container part.</p> <p>6. Apparatus as claimed in claim 5, wherein the stop members are provided with stop surfaces which preferably comprise plastic.</p> <p>7. Apparatus as claimed in claim 5 or 6, wherein the mould container parts are provided on their side walls with stop surfaces which preferably comprise plastic.</p> <p>8. Apparatus as claimed in claim 4, 5, 6 or 7, wherein the edge-forming means comprise an eccentric drive for the frame.</p> <p>9. Apparatus as claimed in any of the foregoing claims, wherein each mould container part is provided with a number of spacer members for supporting the take-off plates at a distance above the green bricks.</p> <p>10. Apparatus as claimed in any of the foregoing claims, wherein the mould container parts are fixed movably onto the conveyor with some clearance in the direction of movement.</p> <p>11. Apparatus as claimed in claim 10, wherein the conveyor is a chain conveyor and the mould container parts are coupled with some clearance in the direction of movement to a chain part connectable to the chain.</p>	

Figure 25

3.1.3. Images inquiry and retrieval

Note, in OPS RESTful services, "images" is the new name for "document (inquiry or retrieval) service" as used in former versions of OPS.

Images inquiry

Use this to find out which images (or "documents") are available for a particular reference.

Request example:

GET <http://ops.epo.org/rest-services/published-data/publication/epodoc/EP1000000.A1/images>

Accept: application/ops+xml

POST <http://ops.epo.org/rest-services/published-data/publication/epodoc/images>

Accept: application/ops+xml

Request Body: EP1000000.A1

Response structure:

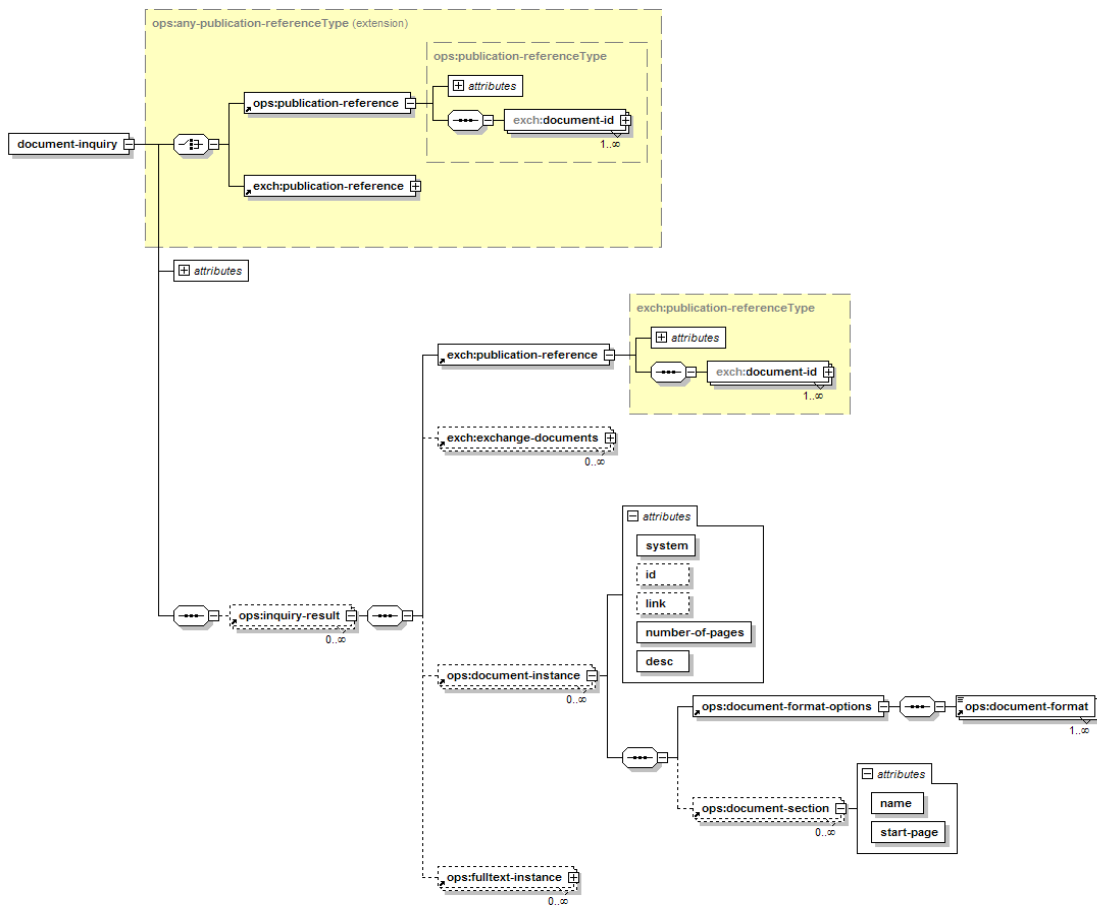


Figure 26

The structure shown in Figure 26 is part of the `ops` namespace and defined in the [ops.xsd](#) schema file.

Response example:

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:document-inquiry>
    <ops:publication-reference>
      <document-id document-id-type="epodoc">
        <doc-number>EP1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:inquiry-result>
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>EP</country>
          <doc-number>1000000</doc-number>
          <kind>A1</kind>
        </document-id>
      </publication-reference>
      <ops:document-instance system="ops.epo.org"
link="EP/1000000/A1/thumbnail" number-of-pages="6" desc="Drawing">
        <ops:document-format-options>
          <ops:document-format>application/pdf</ops:document-format>
          <ops:document-format>application/tiff</ops:document-format>
        </ops:document-format-options>
        <ops:document-section name="DRAWINGS" start-page="1"/>
      </ops:document-instance>
      <ops:document-instance system="ops.epo.org"
link="EP/1000000/A1/fullimage" number-of-pages="12" desc="FullDocument">
        <ops:document-format-options>
          <ops:document-format>application/pdf</ops:document-format>
          <ops:document-format>application/tiff</ops:document-format>
        </ops:document-format-options>
        <ops:document-section name="ABSTRACT" start-page="1"/>
        <ops:document-section name="BIBLIOGRAPHY" start-page="1"/>
        <ops:document-section name="CLAIMS" start-page="3"/>
        <ops:document-section name="DESCRIPTION" start-page="2"/>
        <ops:document-section name="DRAWINGS" start-page="5"/>
        <ops:document-section name="SEARCH_REPORT" start-page="11"/>
      </ops:document-instance>
      <ops:document-instance system="ops.epo.org"
link="EP/1000000/PA/firstpage" number-of-pages="1" desc="FirstPageClipping">
        <ops:document-format-options>
          <ops:document-format>application/pdf</ops:document-format>
          <ops:document-format>application/tiff</ops:document-format>
          <ops:document-format>image/png</ops:document-format>
        </ops:document-format-options>
        <ops:document-section name="ABSTRACT" start-page="1"/>
        <ops:document-section name="BIBLIOGRAPHY" start-page="1"/>
      </ops:document-instance>
    </ops:inquiry-result>
  </ops:document-inquiry>
</ops:world-patent-data>
```

Using the response (see XML above) you can then build the request URI for the Published-data Images retrieval service.

Images retrieval

The first step in the images retrieval is to extract the image references from the inquiry response and construct corresponding URLs, e.g:

- `/rest-services/published-data/images/EP/1000000/PA/firstpage`
- `/rest-services/published-data/images/EP/1000000/A1/thumbnail`
- `/rest-services/published-data/images/EP/1000000/A1/fullimage`

Note, using the response XML copy the `link` attribute of the document instance element:

Retrieve the 1st drawing

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/PA/firstpage
```

```
Accept: image/jpeg
```

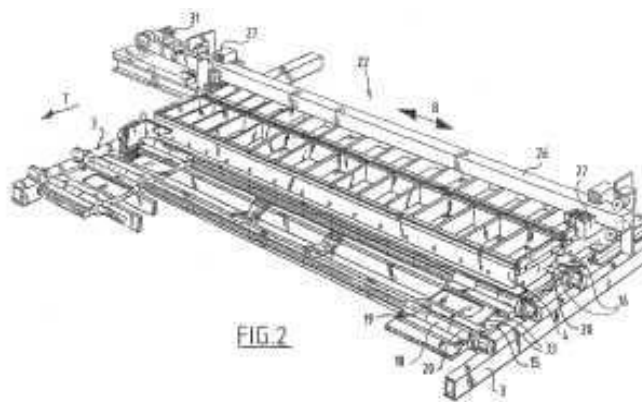


Figure 27: The result of "firstpage.jpeg" request

Accept header is optional and should be set to the value returned in `<ops:document-format>` by [images inquiry](#). By default the first available format is applied. Other examples of the use of Accept header:

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/PA/firstpage
```

```
Accept: image/png
```

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/PA/firstpage
```

```
Accept: image/tiff
```

Alternatively you can specify the type of the image as a resource extension and in that case you don't need to specify the `Accept` header:

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/PA/firstpage.pdf
```

If you need to specify the system different from default `ops.epo.org`, you can do it with a `From` HTTP header or corresponding `From` query parameter. The following two requests are equivalent:

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/PA/firstpage.pdf?From=ops.epo.org
```

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/PA/firstpage
```

```
Accept: application/pdf  
From: ops.epo.org
```

Note, the [firstpage.jpeg](#) request returns JPEG image not wider than 320px. PNG and TIFF images are returned in original size.

Retrieve all images as thumbnails

To retrieve all images as small thumbnails, use the following request:

```
GET http://ops.epo.org/rest-services/published-data/images/EP/1000000/A1/thumbnail
```

```
Accept: application/pdf
```


Accept header is optional and is described above.

X-OPS-Range header or Range query parameter should be provided in order to generate TIFF thumbnail image. Note that only single-page range is supported:

```
GET http://ops.epo.org/rest-services/published-  
data/images/EP/1000000/A1/thumbnail.tiff?Range=4
```

Full document retrieval

Note: In order to provide user with full document, OPS internally must do image inquiry and then assemble full document page by page. It's quite resource consumptive process and might load OPS significantly. Thus it is not possible to get the full document in one request but you can download it page by page. To do this, use the X-OPS-Range HTTP header or Range query parameter to indicate the page you want to retrieve as shown below.

```
GET http://ops.epo.org/rest-services/published-  
data/images/EP/1000000/PA/fullimage  
  
Accept: application/pdf  
X-OPS-Range: 1
```

Accept header is optional and is described above.

X-OPS-Range header is obligatory. It may accept only a single number (not a range). The resulting document consists of a single page (or single image) with a given number as counted in a full document with all pages.

```
GET http://ops.epo.org/rest-services/published-  
data/images/EP/1000000/A1/fullimage.tiff?Range=1
```

Response:

You get the requested page of the full document in PDF format for your request reference (EP1000000.A1):

Response structure:

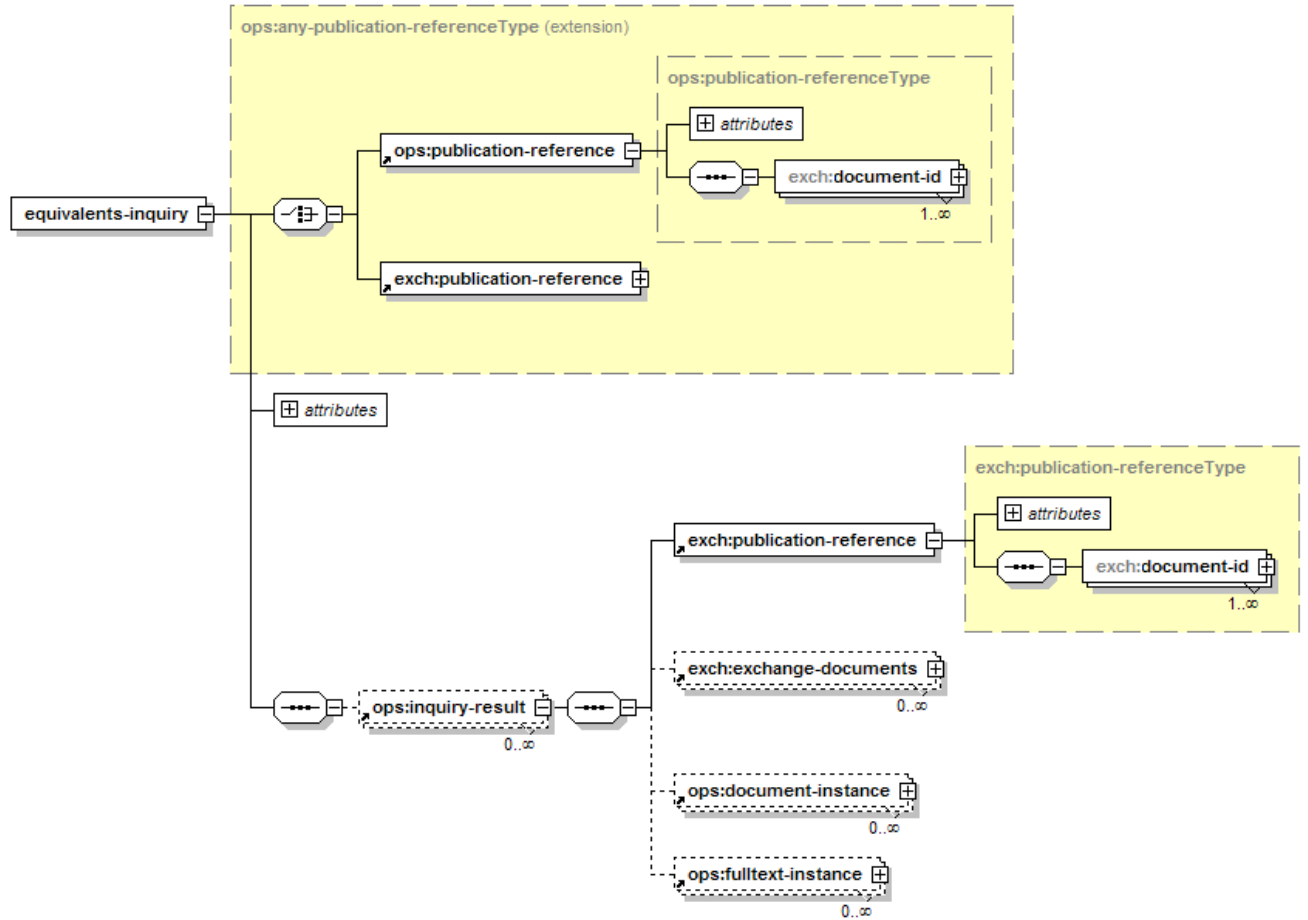


Figure 29

The structure shown in Figure 29 is part of the `ops` namespace and defined in the [ops.xsd](#) schema file.

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:equivalents-inquiry>
    <ops:publication-reference>
      <document-id document-id-type="epodoc">
        <doc-number>EP1000000</doc-number>
      </document-id>
    </ops:publication-reference>
    <ops:inquiry-result>
      <publication-reference>
        <document-id document-id-type="epodoc">
          <doc-number>NL1010536C</doc-number>
        </document-id>
      </publication-reference>
    </ops:inquiry-result>
  </ops:equivalents-inquiry>
</ops:world-patent-data>
```

```

<ops:inquiry-result>
  <publication-reference>
    <document-id document-id-type="epodoc">
      <doc-number>EP1000000</doc-number>
    </document-id>
  </publication-reference>
</ops:inquiry-result>
<ops:inquiry-result>
  <publication-reference>
    <document-id document-id-type="epodoc">
      <doc-number>US6093011</doc-number>
    </document-id>
  </publication-reference>
</ops:inquiry-result>
<ops:inquiry-result>
  <publication-reference>
    <document-id document-id-type="epodoc">
      <doc-number>AT232441T</doc-number>
    </document-id>
  </publication-reference>
</ops:inquiry-result>
</ops:equivalents-inquiry>
</ops:world-patent-data>

```

Note, when requesting equivalents it is also possible to retrieve a combination (abstract, biblio, full-cycle, and/or images.) The URI is constructed by placing the constituent(s) after /equivalents separating them with commas. (See examples below.)

Equivalents with abstract:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000/equivalents/abstract
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/equivalents/abstract
```

```
Accept: application/ops+xml
```

```
Request Body: EP1000000
```

Response:

```

<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:equivalents-inquiry>

```

```

<ops:publication-reference>
  <document-id document-id-type="epodoc">
    <doc-number>EP1000000</doc-number>
  </document-id>
</ops:publication-reference>
<ops:inquiry-result>
  <publication-reference>
    <document-id document-id-type="epodoc">
      <doc-number>EP1000000</doc-number>
    </document-id>
  </publication-reference>
  <exchange-documents>
    <exchange-document system="ops.epo.org" country="EP" doc-
number="1000000" kind="A1">
      <bibliographic-data>
        <publication-reference>
          <document-id document-id-type="docdb">
            <country>EP</country>
            <doc-number>1000000</doc-number>
            <kind>A1</kind>
            <date>20000517</date>
          </document-id>
          <document-id document-id-type="epodoc">
            <doc-number>EP1000000</doc-number>
            <date>20000517</date>
          </document-id>
        </publication-reference>
        <parties/>
      </bibliographic-data>
      <abstract lang="en">
        <p>The invention relates to an apparatus (1) for manufacturing
green bricks from clay for the brick manufacturing industry, comprising a
circulating conveyor (3) carrying mould containers combined to mould
container parts (4), a reservoir (5) for clay arranged above the mould
containers, means for carrying clay out of the reservoir (5) into the mould
containers, means (9) for pressing and trimming clay in the mould
containers, means (11) for supplying and placing take-off plates for the
green bricks (13) and means for discharging green bricks released from the
mould containers, characterized in that the apparatus further comprises
means (22) for moving the mould container parts (4) filled with green bricks
such that a protruding edge is formed on at least one side of the green
bricks. &lt;IMAGE></p>
      </abstract>
    </exchange-document>
  </exchange-documents>
</ops:inquiry-result>
<!-- remaining inquiry results removed for readability -->
</ops:equivalents-inquiry>
</ops:world-patent-data>

```

Equivalents and biblio:

GET <http://ops.epo.org/rest-services/published-data/publication/epodoc/EP1000000.A1/equivalents/biblio>

Accept: application/ops+xml

POST <http://ops.epo.org/rest-services/published-data/publication/epodoc/equivalents/biblio>

Accept: application/ops+xml

Request Body: EP1000000.A1

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:equivalents-inquiry>
    <ops:publication-reference>
      <document-id document-id-type="epodoc">
        <doc-number>EP1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:inquiry-result>
      <publication-reference>
        <document-id document-id-type="epodoc">
          <doc-number>NL1010536C</doc-number>
        </document-id>
      </publication-reference>
      <exchange-documents>
        <exchange-document system="ops.epo.org" family-id="19768124"
country="NL" doc-number="1010536" kind="C2">
          <bibliographic-data>
            <publication-reference>
              <document-id document-id-type="docdb">
                <country>NL</country>
                <doc-number>1010536</doc-number>
                <kind>C2</kind>
                <date>20000515</date>
              </document-id>
              <document-id document-id-type="epodoc">
                <doc-number>NL1010536C</doc-number>
                <date>20000515</date>
              </document-id>
            </publication-reference>
            <!-- classifications removed for readability -->
            <application-reference doc-id="41488391">
              <document-id document-id-type="epodoc">
                <doc-number>NL19981010536</doc-number>
                <date>19981112</date>
              </document-id>
              <document-id document-id-type="original">
```

```

        <doc-number>1010536</doc-number>
    </document-id>
</application-reference>
    <!-- claims, parties, title and citations removed for readability -->
</bibliographic-data>
</exchange-document>
</exchange-documents>
</ops:inquiry-result>
    <!-- remaining inquiry results removed for readability -->
</ops:equivalents-inquiry>
</ops:world-patent-data>

```

Equivalents and biblio and full-cycle:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000.A1/equivalents/biblio,full-cycle
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/equivalents/biblio,full-cycle
```

```
Accept: application/ops+xml  
Request Body: EP1000000.A1
```

Response:

```

<ops:inquiry-result>
  <publication-reference>
    <document-id document-id-type="epodoc">
      <doc-number>EP1000000</doc-number>
    </document-id>
  </publication-reference>
  <exchange-documents>
    <exchange-document system="ops.epo.org" family-id="19768124"  
country="EP" doc-number="1000000" kind="A1">
      <bibliographic-data>
        <publication-reference>
          <document-id document-id-type="docdb">
            <country>EP</country>
            <doc-number>1000000</doc-number>
            <kind>A1</kind>
            <date>20000517</date>
          </document-id>
          <document-id document-id-type="epodoc">
            <doc-number>EP1000000</doc-number>
            <date>20000517</date>
          </document-id>
        </publication-reference>

```

```
        <!-- remaining bibliographic data removed -->
    </exchange-document>
    <exchange-document system="ops.epo.org" family-id="19768124"
country="EP" doc-number="1000000" kind="B1">
    <bibliographic-data>
    <publication-reference>
    <document-id document-id-type="docdb">
    <country>EP</country>
    <doc-number>1000000</doc-number>
    <kind>B1</kind>
    <date>20030212</date>
    </document-id>
    <document-id document-id-type="epodoc">
    <doc-number>EP1000000</doc-number>
    <date>20030212</date>
    </document-id>
    </publication-reference>
    <!-- remaining bibliographic data removed -->
    </bibliographic-data>
    </exchange-document>
</exchange-documents>
</ops:inquiry-result>
```

Equivalents and images:

```
GET http://ops.epo.org/rest-services/published-  
data/publication/epodoc/EP1000000.A1/equivalents/images
```

```
Accept: application/exchange+xml
```

```
POST http://ops.epo.org/rest-services/published-  
data/publication/epodoc/equivalents/images
```

```
Accept: application/exchange+xml
```

```
Request Body: EP1000000.A1
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:equivalents-inquiry>
    <ops:publication-reference>
      <document-id document-id-type="epodoc">
        <doc-number>EP1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:inquiry-result>
      <publication-reference>
        <document-id document-id-type="epodoc">
          <doc-number>NL1010536C</doc-number>
        </document-id>
      </publication-reference>
      <ops:document-instance system="ops.epo.org"
link="NL/1010536/C2/thumbnail" number-of-pages="6" desc="Drawing">
        <ops:document-format-options>
          <ops:document-format>application/pdf</ops:document-format>
          <ops:document-format>application/tiff</ops:document-format>
        </ops:document-format-options>
        <ops:document-section name="DRAWINGS" start-page="1"/>
      </ops:document-instance>
      <ops:document-instance system="ops.epo.org"
link="NL/1010536/C2/fullimage" number-of-pages="19" desc="FullDocument">
        <ops:document-format-options>
          <ops:document-format>application/pdf</ops:document-format>
          <ops:document-format>application/tiff</ops:document-format>
        </ops:document-format-options>
        <ops:document-section name="ABSTRACT" start-page="1"/>
        <ops:document-section name="BIBLIOGRAPHY" start-page="1"/>
        <ops:document-section name="CLAIMS" start-page="9"/>
        <ops:document-section name="DESCRIPTION" start-page="2"/>
        <ops:document-section name="DRAWINGS" start-page="11"/>
      </ops:document-instance>
    </ops:inquiry-result>
  </ops:equivalents-inquiry>
</ops:world-patent-data>
```



```
<ops:document-section name="SEARCH_REPORT" start-page="18"/>
</ops:document-instance>
</ops:inquiry-result>
<!-- remaining results removed for readability -->
</ops:equivalents-inquiry>
</ops:world-patent-data>
```

- 3.1. **Published-data services**
- 3.2. **Family service**
- 3.3. **Number-service**
- 3.4. **Register service**
- 3.5. **Legal service**
- 3.6. **Classification services**

[top of document](#)

3.2. Family service

The family service retrieves relatives of a [patent family](#), essentially the patent numbers related to your input. The family retrieved is the INPADOC extended patent family (see [the espacenet help](#) for more details)

As you have seen [input formats](#) used in patent information have a variety of purposes. Relevant to the family service, requests can be either docdb or epodoc number format, but not original.

With the family service you can also modify your responses with 'constituents' *biblio*, *legal* (or none). Constituents of the family service are data blocks that attribute to a specific patent family.

Constituent	Description
<i>biblio</i>	Delivers bibliographic data of each family member in the exchange-document element
<i>legal</i>	Includes legal codes and descriptions of the family members

Table 20

Generic request structure:

```
GET http://ops.epo.org/rest-services/family/[reference-type:
publication OR application OR priority]/[input-format: docdb OR
epodoc]/[input e.g. EP1000000 OR EP1000000.A1]/[constituent: e.g.
biblio,legal]

Accept: application/ops+xml
```

In the request structure, **bold means mandatory**, *italics means optional*.

Note, the family service supports the * **wildcard** that can be used for one letter of the given kind code; e.g. A*(matches A1, A2,..), * (matches every kind code).

Generic request example:

```
GET http://ops.epo.org/rest-services/family/publication/docdb/EP.1000000.A1
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-services/family/  
publication/docdb
```

```
Accept: application/ops+xml
```

```
Request Body: EP.1000000.A1
```

Note, specific request and response examples are given in the following sections.

Generic response structure

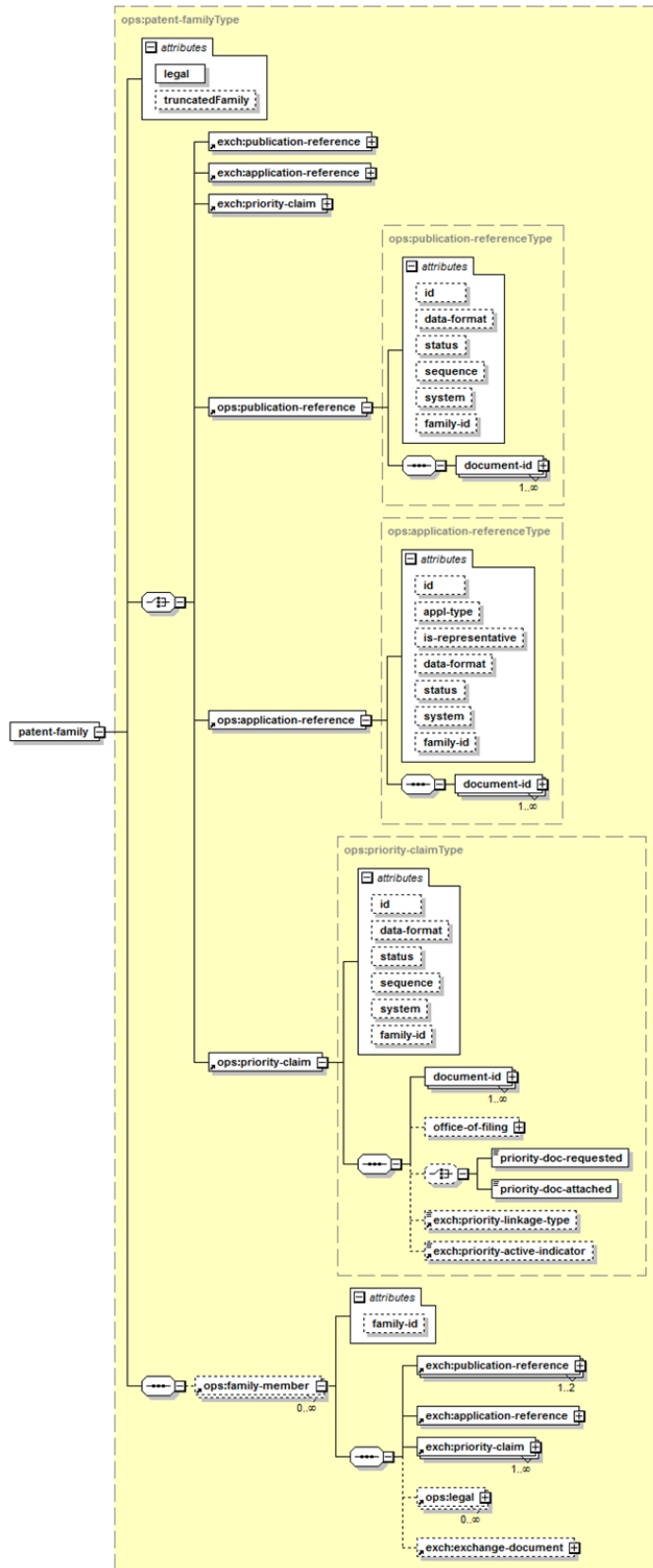


Figure 30

The structure shown in Figure 30 is part of the `ops` namespace and defined in the [ops.xsd](#) schema file.

3.2.1. Family data (without constituents)

```
GET http://ops.epo.org/rest-services/family/publication/docdb/EP.1000000.A1
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-services/family/publication/docdb
```

```
Accept: application/ops+xml
```

```
Request Body: EP.1000000.A1
```

Response:

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:patent-family legal="false" total-result-count="6">
    <ops:publication-reference>
      <document-id document-id-type="docdb">
        <country>EP</country>
        <doc-number>1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:family-member>
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>AT</country>
          <doc-number>232441</doc-number>
          <kind>T</kind>
          <date>20030215</date>
        </document-id>
        <document-id document-id-type="epodoc">
          <doc-number>AT232441T</doc-number>
          <date>20030215</date>
        </document-id>
      </publication-reference>
      <application-reference doc-id="1370469">
        <document-id document-id-type="docdb">
          <country>AT</country>
          <doc-number>99203729</doc-number>
          <kind>T</kind>
          <date>19991108</date>
        </document-id>
      </application-reference>
      <priority-claim kind="national">
        <document-id document-id-type="docdb">
          <country>NL</country>
          <doc-number>1010536</doc-number>
```

```

        <kind>A</kind>
        <date>19981112</date>
    </document-id>
    </priority-claim>
</ops:family-member>
<!-- Remaining 5 family members removed for readability of the example -->
</ops:patent-family>
</ops:world-patent-data>

```

3.2.2. Family data with biblio

Request:

```
GET http://ops.epo.org/rest-
services/family/publication/docdb/EP.1000000.A1/biblio
```

```
Accept:application/ops+xml
```

```
POST http://ops.epo.org/rest-
services/family/publication/docdb/biblio
```

```
Accept:application/ops+xml
Request Body: EP.1000000.A1
```

Response:

```

<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:patent-family legal="false" total-result-count="6">
    <ops:publication-reference>
      <document-id document-id-type="docdb">
        <country>EP</country>
        <doc-number>1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:family-member>
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>AT</country>
          <doc-number>232441</doc-number>
          <kind>T</kind>
          <date>20030215</date>
        </document-id>
        <document-id document-id-type="epodoc">
          <doc-number>AT232441T</doc-number>
          <date>20030215</date>
        </document-id>
      </publication-reference>

```

```

<application-reference>
  <document-id document-id-type="docdb">
    <country>AT</country>
    <doc-number>99203729</doc-number>
    <kind>T</kind>
    <date>19991108</date>
  </document-id>
</application-reference>
<priority-claim kind="national">
  <document-id document-id-type="docdb">
    <country>NL</country>
    <doc-number>1010536</doc-number>
    <kind>A</kind>
    <date>19981112</date>
  </document-id>
</priority-claim>
<exchange-document system="ops.epo.org" family-id="19768124"
country="AT" doc-number="232441" kind="T">
  <bibliographic-data>
    <publication-reference>
      <document-id document-id-type="docdb">
        <country>AT</country>
        <doc-number>232441</doc-number>
        <kind>T</kind>
        <date>20030215</date>
      </document-id>
      <document-id document-id-type="epodoc">
        <doc-number>AT232441T</doc-number>
        <date>20030215</date>
      </document-id>
    </publication-reference>
    <!-- rest of the bibliographic data removed for readability -->
  </bibliographic-data>
</exchange-document>
</ops:family-member>
<!-- more family members here -->
</ops:patent-family>
</ops:world-patent-data>

```

3.2.3. Family data with legal

Request:

```
GET http://ops.epo.org/rest-
services/family/publication/docdb/EP.1000000.A1/legal
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-
services/family/publication/docdb/legal
```

Accept: application/ops+xml
Request Body: EP.1000000.A1

Response:

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:patent-family legal="true" total-result-count="6">
    <ops:publication-reference>
      <document-id document-id-type="docdb">
        <country>EP</country>
        <doc-number>1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:family-member>
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>AT</country>
          <doc-number>232441</doc-number>
          <kind>T</kind>
          <date>20030215</date>
        </document-id>
        <document-id document-id-type="epodoc">
          <doc-number>AT232441T</doc-number>
          <date>20030215</date>
        </document-id>
      </publication-reference>
      <application-reference>
        <document-id document-id-type="docdb">
          <country>AT</country>
          <doc-number>99203729</doc-number>
          <kind>T</kind>
          <date>19991108</date>
        </document-id>
      </application-reference>
      <priority-claim kind="national">
        <document-id document-id-type="docdb">
          <country>NL</country>
          <doc-number>1010536</doc-number>
          <kind>A</kind>
          <date>19981112</date>
        </document-id>
      </priority-claim>
      <priority-active-indicator>YES</priority-active-indicator>
    </ops:family-member>
    <ops:legal code="RER"
      desc="CEASED AS TO PARAGRAPH 5 LIT. 3 LAW INTRODUCING
      PATENT TREATIES"
      infl="-"
      dateMigr="00010101">
      <ops:pre line="00001">AT 99203729T 2003-08-15RER -CEASED AS TO
      PARAGRAPH 5 LIT. 3 LAW INTRODUCING PATENT TREATIES</ops:pre>
      <ops:L001EP desc="Country Code">AT</ops:L001EP>
      <ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
      <ops:L003EP desc="Document Number"> 99203729</ops:L003EP>
    </ops:legal>
  </ops:patent-family>
</ops:world-patent-data>
```



```

    <ops:L004EP desc="Kind Code">T</ops:L004EP>
    <ops:L005EP desc="IPR Type">PI</ops:L005EP>
    <ops:L006EP desc="PRS DOCUMENT TYPE">P</ops:L006EP>
    <ops:L007EP desc="Gazette DATE">2003-08-15</ops:L007EP>
    <ops:L008EP desc="Legal Event Code 1">RER</ops:L008EP>
    <ops:L018EP desc="DATE last exchanged">2003-08-28</ops:L018EP>
    <ops:L019EP desc="DATE first created">2003-08-26</ops:L019EP>
    <ops:L500EP/>
  </ops:legal>
</ops:family-member>
  <!-- more family members here -->
</ops:patent-family>
</ops:world-patent-data>

```

3.2.4. Family data with biblio and legal

Request:

```
GET http://ops.epo.org/rest-services/family/publication/docdb/EP.1000000.A1/biblio,legal
```

```
Accept:application/ops+xml
```

```
POST http://ops.epo.org/rest-services/family/publication/docdb/biblio,legal
```

```
Accept:application/ops+xml
Request Body: EP.1000000.A1
```

Response:

```

<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:patent-family legal="true" total-result-count="6">
    <ops:publication-reference>
      <document-id document-id-type="docdb">
        <country>EP</country>
        <doc-number>1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:family-member>
      <!-- references removed for readability, compare previous family examples -->
      <ops:legal code="RER" desc="CEASED AS TO PARAGRAPH 5 LIT. 3 LAW INTRODUCING PATENT TREATIES" infl="-" dateMigr="00010101">
        <ops:pre line="00001">AT 99203729T 2003-08-15RER -CEASED AS TO PARAGRAPH 5 LIT. 3 LAW INTRODUCING PATENT TREATIES</ops:pre>

```

```

<ops:L001EP desc="Country Code">AT</ops:L001EP>
<ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
<ops:L003EP desc="Document Number"> 99203729</ops:L003EP>
<ops:L004EP desc="Kind Code">T</ops:L004EP>
<ops:L005EP desc="IPR Type">PI</ops:L005EP>
<ops:L006EP desc="PRS DOCUMENT TYPE">P</ops:L006EP>
<ops:L007EP desc="Gazette DATE">2003-08-15</ops:L007EP>
<ops:L008EP desc="Legal Event Code 1">RER</ops:L008EP>
<ops:L018EP desc="DATE last exchanged">2003-08-28</ops:L018EP>
<ops:L019EP desc="DATE first created">2003-08-26</ops:L019EP>
<ops:L500EP/>
</ops:legal>
<exchange-document system="ops.epo.org" family-id="19768124"
country="AT" doc-number="232441" kind="T">
  <bibliographic-data>
    <publication-reference>
      <document-id document-id-type="docdb">
        <country>AT</country>
        <doc-number>232441</doc-number>
        <kind>T</kind>
        <date>20030215</date>
      </document-id>
      <document-id document-id-type="epodoc">
        <doc-number>AT232441T</doc-number>
        <date>20030215</date>
      </document-id>
    </publication-reference>
    <!-- remaining bibliographic data removed -->
  </bibliographic-data>
</exchange-document>
</ops:family-member>
<!-- more family members here -->
</ops:patent-family>
</ops:world-patent-data>

```

INPADOC family is too big to include in the response

In situations when the requested family is very large (several hundred members), all of the members cannot be included in the response due to technical reasons. The patent-family structure is returned with attribute `truncatedFamily="true"` and only a limited number of family members are returned. Shown below is a family **response example** for publication US2003082717 including the first four family members:

Patent Family		legal: false - truncatedFamily: true
Meta:	Elapsed Time: 3363	<ops:document-i number-of-pages
Publication Reference:	US2003082717	
Family Member (1):	US 2003082717 A1 20030501 US2003082717 20030501 US 18301602 A 20020626	
Family Member (2):	AR 064032 A1 20090304 AR064032 20090304 AR P070105308 A 20071129	
Family Member (3):	AT 268819 T 20040615 AT268819T 20040615 AT 01126188 T 19981201	
Family Member (4):	AT 270675 T 20040715 AT270675T 20040715 AT 01126187 T 19981201	

Figure 31

- 3.1. **Published-data services**
- 3.2. **Family service**
- 3.3. **Number-service**
- 3.4. **Register service**
- 3.5. **Legal service**
- 3.6. **Classification services**

[top of document](#)

3.3. Number-service

This service converts numbers from one [input format](#) into another format. Number conversion always pertains to the same [reference](#) (publication, application or priority.)

List of possible conversions:

Input format	Output format
original	docdb, epodoc
docdb	epodoc, original
epodoc	original

Table 21

Note:

The number-service uses a carefully crafted set of pattern matching and formatting rules for number conversion based on the OPS [input formats](#). The development of these rules is an ongoing process and **at this point in time not every possible format of every single country is supported**.

The rules are **optimized for recent documents of big countries and automated data exchange**. However, the rules are **constantly updated** and thus a conversion that leads to a wrong result today might work next week. It is important to understand that all requests will lead to a conversion but that the quality of the result will be variable.

Even though it is possible to request the number-service with a single string, it is strongly advised to provide the date, country code and kind code separately with your `input` (number) whenever you can identify them. It is especially important to **include the date** in number-service requests because number formatting changes over time.

Status codes

In number-service responses you will find an `ops:meta` element that contains status information regarding the number conversion. If you experience problems with the number-service, including the status codes helps the support team to analyse them.

Message Code	Message	Type
pBRE001	Check digit not correct	ERROR
pBRE002	No country code provided	WARNING
pBRE003	Kind code not found	WARNING
pBRE004	Date out of range	WARNING
pBRE005	No number format detected for this kind code	WARNING
pBRE006	Number transformation rule not specified	WARNING
pBRE007	Number outside number range	WARNING
pBRE008	Converted to Gregorian year	WARNING
pBRE009	Converted to Emperor year	WARNING
pBRE010	Date generated through lookup with input number	WARNING
pBRE011	No match found. Country code changed.	WARNING
pBRE012	Kind code in number does not match kind code supplied.	ERROR
pBRE013	Output kind code different from extracted kind code.	WARNING
pBRE014	Kind code generated.	WARNING
pBRE015	Kind code transformed.	WARNING
pBRE016	Provided date is different from extracted date.	WARNING
pBRE017	Publication date set from extracted date.	WARNING
pBRE018	Application date set from extracted date.	WARNING
pBRE019	No kind code provided, setting it to A.	WARNING
pBRE020	Kind code does not start with a letter, setting it to A.	WARNING
pBRE021	Second character of provided kind code is not a digit, removing it..	WARNING
pBRE022	Date is before year 1617	ERROR
pBRE023	Date is after current year + 2	ERROR
pBRE024	Month is not between 1 and 12, month and day set to 0	ERROR
pBRE025	Day is invalid, set to 0	ERROR
pBRE026	Date is in the future.	WARNING
pBRE027	Date lookup with input number did not find anything.	ERROR
pBRE028	Date generated through lookup with output number	WARNING
pBRE029	Date lookup with output number did not find	ERROR

	anything.	
pBRE030	Document with kind code found in master database.	WARNING
pBRE031	Kind code updated via lookup.	WARNING
pBRE032	Kind code lookup did not find anything.	WARNING
pBRE033	REFI kind code lookup did not find anything.	WARNING
pBRE034	REFI not allowed.	WARNING
pBRE035	Citing target set.	WARNING
pBRE036	Citing target not set, kind codes do not match.	WARNING
pBRE037	Date corrected via kind code lookup.	WARNING
pBRE038	Date set via kind code lookup.	WARNING
pBRE039	No Active Priority	WARNING
pBRE040	Error in refi kind code lookup.	ERROR
pBRE041	Last change origin of targetId was human, setting targetId as rawId.	WARNING
pBRE042	The provided document is a valid DOCDB format.	WARNING
pBRE043	The provided document is not a valid DOCDB format.	ERROR
pBRE044	Rules without country used for transforming the number.	WARNING
pBRE045	Provided country code is different than extracted country code.	WARNING
pBRE046	Kind code rule did not allow configured rules to transform number.	ERROR
pBRE047	Meaningless digits removed from number	WARNING
pBRE048	Date generated through sequence number lookup	WARNING
pBRE049	Date lookup with sequence did not find anything	WARNING
pBRE050	RID found in master database	WARNING
pBRE051	RID not found in master database	WARNING
pBRE052	Document found by RID in master database	WARNING
pBRE053	Document not found by RID in master database	WARNING
pBRE054	Error in RID lookup	WARNING
pBRE055	Document found in master database	WARNING
pBRE056	Document not found in master database	WARNING
pBRE057	preBRE not configured for validation	WARNING
pBRE058	Kind code is missing for validation	WARNING
pBRE059	Date is missing for validation	WARNING
pBRE060	Country code is missing for validation	ERROR
pBRE061	Country code is invalid for validation	ERROR
pBRE100	Duplicate priority removed	WARNING
pBRE101	Country code standardised	WARNING
pBRE102	Leading/trailing characters removed	WARNING
pBRE103	Country code removed for valid PCT number format	WARNING

pBRE104	Chinese check digit removed	WARNING
pBRE105	Country code derived from serial number	WARNING
pBRE106	Country code EP/EU changed to EM	WARNING
pBRE107	Duplicate country code removed	WARNING
pBRE108	Country code changed due to date	WARNING
pBRE109	Country code derived from serial number	WARNING
pBRE110	Kind code set to W for PCT protection type	WARNING
pBRE111	Kind code set to W for Epoque PCT priority	WARNING
pBRE112	EP-A priority created from self-claimer	WARNING
pBRE113	Kind code set to W for protection type PCT	WARNING
pBRE114	LMI set for PCT priority	WARNING
pBRE115	Country code set to WO for Epoque format	WARNING
pBRE116	Country code removed for external PCT format	WARNING
pBRE117	EP priority generated for HU translation	WARNING
pBRE118	LMI Removed	WARNING
pBRE119	Kind code W forced to use the no country rules	WARNING
pBRE120	Kind code set to W for application associated with publication country WO	WARNING
pBRE121	Country code for priority starting with DM changed to XH	WARNING
pBRE122	new country code set for KR citation	WARNING
pBRE123	Kind code derived from serial number	WARNING
pBRE124	Italian priority selected from multiple IT priorities provided	WARNING
pBRE126	Kind code derived from number	WARNING
pBRE127	Kind code removed from number	WARNING
pBRE128	JP single emperor year prepended with 0	WARNING
pBRE129	Application kind code set to U for utility model publication	WARNING
pBRE130	Application kind code set to T for national translation of EP	WARNING
pBRE131	Spaces before/after separator removed	WARNING
pBRE132	Leading zero added for single digit years (2000-2009)	WARNING
pBRE133	Substitution of Chinese comma by a regular comma	WARNING
pBRE134	New kind code U set for KR citation	WARNING
pBRE135	New raw ID set for KR citation	WARNING
pBRE136	New kind code set for KR citation	WARNING
pBRE137	Transformed small caps characters to uppercase	WARNING
pBRE138	Country code set to IB for DOCDB format	WARNING
pBRE139	Priority removed	WARNING
pBRE999	Number could not be transformed, setting number from original	ERROR

Table 22

Message Code	Message
BRE000	Unknown error code
BRE001	Number Standardization Stopped
BRE002	Number Standardization Stopped - Routed to Post Processing
BRE003	RawId not Defined
BRE004	Country Code not Valid
BRE005	Exception in date lookup
BRE006	No Matching Pattern Found
BRE007	Failed to convert sequence number for document
BRE008	Illegal Country Code WO for APP-PRI
BRE009	Invalid Sequence Number in KindCodeSet
BRE010	Invalid Sequence Number in GenerateTargetId
BRE011	Invalid Length for SN
BRE012	SN Length Over Max Size
BRE013	Kind Code not valid for this Time Period
BRE014	Attempt to Apply Pattern that was not Parsed
BRE015	Failed to Convert Year Group
BRE016	Failed to Convert Sequence Number Group
BRW001	Country Code not Defined
BRW002	Document Date not Defined
BRW003	Kind Code not Defined
BRW004	Illegal Country Code for APP-PRI
BRW005	DM found in raw id, using XH country
BRW006	Using Default Country Pattern
BRW007	Country Code Replaced
BRW008	Year indication from date field
BRW009	Sequence number adjusted to
BRW010	Kind Code Relation not Allowed
BRW011	Failed to Define TargetId. RawId used instead
BRW012	Failed to Define TargetId. RawId used instead
BRW013	Failed to Define TargetId. RawId used instead
BRW014	No Kind Code or Generation Format Defined
BRW015	No Date Found. Trigger lookup
BRW016	Date Lookup Did not Find Anything
BRW017	Year Replacement Failed. No Date Defined in the Input
BRW018	Year Defined in Target Format, but no Date Defined
BRW019	Document Found in DB
BRW020	Kind code replaced by preferred lookup value
BRW021	Kind code replaced but not with first preferred lookup value
BRW022	Lookup found values which are not in preferred list
BRW023	Kind code lookup did not find anything

BRW024	Lookup for kind codes with date contains more than one value
BRW026	Lookup found documents which are not in preferred list
BRW027	Invalid date format
BRW028	Invalid date format
BRW030	Kind code lookup without date found values
BRW031	No Kindcode format defined, kindcode first char matches a valid kindcode
BRW032	No Kindcode format defined, kindcode matches a valid kindcode

Table 23

Generic request structure:

```
GET http://ops.epo.org/rest-services/[service]/[reference-type]/input-format: docdb OR epodoc OR original -- which ever format corresponds with your input =>/input: e.g. EP1000000.A1 OR EP.1000000.* OR BY1%20202345.*.20090622/output-format: docdb OR epodoc OR original

Accept: application/ops+xml
```

In the request structure, **bold means mandatory**.

Generic request example:

```
GET http://ops.epo.org/rest-services/number-service/publication/docdb/EP.1000000/epodoc
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-services/number-service/publication/docdb/epodoc
```

```
Accept: application/ops+xml
```

```
Request Body: EP.1000000
```


Generic response structure:

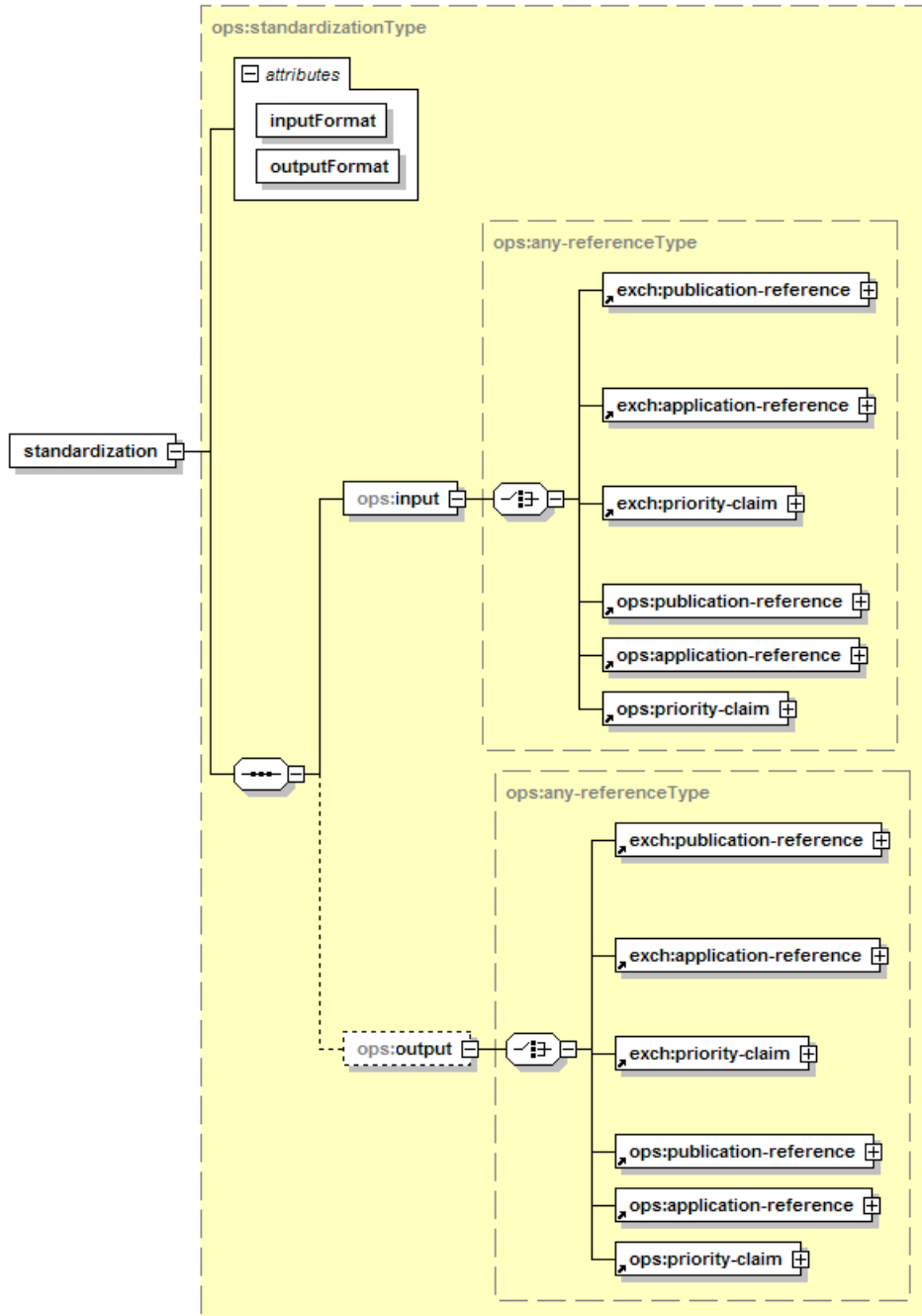


Figure 32

The structure shown in Figure 32 is part of the `ops` namespace and defined in the [ops.xsd](#) schema file.

Below you will find various number-service **request examples and responses**.

3.3.1.From docdb to epodoc

```
GET http://ops.epo.org/rest-services/number-service/application/docdb/MD.20050130.A.20050130/epodoc
```

```
Accept: application/ops+xml
```

```
POST http://ops.epo.org/rest-services/number-service/application/docdb/epodoc
```

```
Accept: application/ops+xml
```

```
Request Body: MD.20050130.A.20050130
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="/style/numbers.xsl"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:meta name="version:docdb/epodoc" value="2010.03.19"/>
  <ops:standardization inputFormat="docdb" outputFormat="epodoc">
    <ops:input>
      <ops:application-reference>
        <document-id document-id-type="docdb">
          <country>MD</country>
          <doc-number>20050130</doc-number>
          <kind>A</kind>
          <date>20050130</date>
        </document-id>
      </ops:application-reference>
    </ops:input>
    <ops:output>
      <ops:application-reference>
        <document-id document-id-type="epodoc">
          <country>MD</country>
          <doc-number>20050000130</doc-number>
          <date>20050130</date>
        </document-id>
      </ops:application-reference>
    </ops:output>
  </ops:standardization>
</ops:world-patent-data>
```

3.3.2.From original to docdb (JPO example)

```
GET http://ops.epo.org/rest-services/number-service/application/original/JP.\(2006-147056\).A.20060526/docdb
```

Accept: application/ops+xml

POST <http://ops.epo.org/rest-services/number-service/application/original/docdb>

Accept: application/ops+xml

Request Body: **JP.2006-147056.A.20060526**

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="/style/numbers.xsl"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:meta name="status" value="SUCCESS"/>
  <ops:meta name="version" value="10.09.93"/>
  <ops:standardization inputFormat="original" outputFormat="docdb">
    <ops:input>
      <ops:application-reference>
        <document-id document-id-type="original">
          <country>JP</country>
          <doc-number>2006-147056</doc-number>
          <kind>A</kind>
          <date>20060526</date>
        </document-id>
      </ops:application-reference>
    </ops:input>
    <ops:output>
      <ops:application-reference>
        <document-id document-id-type="docdb">
          <country>JP</country>
          <doc-number>2006147056</doc-number>
          <kind>A</kind>
          <date>20060526</date>
        </document-id>
      </ops:application-reference>
    </ops:output>
  </ops:standardization>
</ops:world-patent-data>
```

3.3.3.From docdb to original (JPO example)

GET <http://ops.epo.org/rest-services/number-service/application/docdb/JP.2006147056.A.20060526/original>

Accept: application/ops+xml

POST <http://ops.epo.org/rest-services/number-service/application/docdb/original>

Accept: application/ops+xml

Request Body: **JP.2006147056.A.20060526**

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="/style/numbers.xsl"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:meta name="status" value="SUCCESS"/>
  <ops:meta name="version" value="10.09.93"/>
  <ops:standardization inputFormat="docdb" outputFormat="original">
    <ops:input>
      <ops:application-reference>
        <document-id document-id-type="docdb">
          <country>JP</country>
          <doc-number>2006147056</doc-number>
          <kind>A</kind>
          <date>20060526</date>
        </document-id>
      </ops:application-reference>
    </ops:input>
    <ops:output>
      <ops:application-reference>
        <document-id document-id-type="original">
          <country>JP</country>
          <doc-number>2006-147056</doc-number>
          <kind>A</kind>
          <date>20060526</date>
        </document-id>
      </ops:application-reference>
    </ops:output>
  </ops:standardization>
</ops:world-patent-data>
```

3.3.4.From original to epodoc (USPTO example)

GET [http://ops.epo.org/rest-services/number-service/application/original/US.\(08/921,321\).A.19970829/epodoc](http://ops.epo.org/rest-services/number-service/application/original/US.(08/921,321).A.19970829/epodoc)

Accept: application/ops+xml

POST <http://ops.epo.org/rest-services/number-service/application/original/epodoc>

Accept: application/ops+xml

Request Body: **US . (08/921 , 321) .A.19970829**

Note, in this case it is not possible to remove the brackets in the POST request because of the comma that is part of the input but is also used to separate input references in the request body.

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="/style/numbers.xsl"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:meta name="version:docdb/epodoc" value="2010.03.19"/>
  <ops:meta name="version:original/docdb" value="10.09.93"/>
  <ops:standardization inputFormat="original" outputFormat="epodoc">
    <ops:input>
      <ops:application-reference>
        <document-id document-id-type="original">
          <country>US</country>
          <doc-number>08/921,321</doc-number>
          <kind>A</kind>
          <date>19970829</date>
        </document-id>
      </ops:application-reference>
    </ops:input>
    <ops:output>
      <ops:application-reference>
        <document-id document-id-type="epodoc">
          <country>US</country>
          <doc-number>19970921321</doc-number>
          <date>19970829</date>
        </document-id>
      </ops:application-reference>
    </ops:output>
  </ops:standardization>
</ops:world-patent-data>
```

3.3.5.From epodoc to original

For certain authorities, the conversion of application/priority numbers into DOCDB or EPODOC formats may lead to loss of essential information necessary to convert back these numbers into the original format. For this reason we suggest to use the conversion service from DOCDB or EPODOC into original format for application/priority numbers with caution as it may lead to erroneous results.

3.3.6.From original to docdb (PCT example)

GET [http://ops.epo.org/rest-services/number-service/application/original/\(PCT/GB02/04635\).20021011/docdb](http://ops.epo.org/rest-services/number-service/application/original/(PCT/GB02/04635).20021011/docdb)

Accept: application/ops+xml

POST <http://ops.epo.org/rest-services/number-service/application/original/docdb>

Accept: application/ops+xml

Request Body: **PCT/GB02/04635.20021011**

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="/style/numbers.xsl"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:meta name="status" value="BRW001 BRW003 BRW007 BRW006"/>
  <ops:meta name="version" value="10.09.93"/>
  <ops:standardization inputFormat="original" outputFormat="docdb">
    <ops:input>
      <ops:application-reference>
        <document-id document-id-type="original">
          <doc-number>PCT/GB02/04635</doc-number>
          <date>20021011</date>
        </document-id>
      </ops:application-reference>
    </ops:input>
    <ops:output>
      <ops:application-reference>
        <document-id document-id-type="docdb">
          <country>GB</country>
          <doc-number>0204635</doc-number>
          <kind>W</kind>
          <date>20021011</date>
        </document-id>
      </ops:application-reference>
    </ops:output>
  </ops:standardization>
</ops:world-patent-data>
```

Note, please refer to [section 4.4 Expressing PCT application numbers in docdb format](#) for further information on the PCT format.

- 3.1. **Published-data services**
- 3.2. **Family service**
- 3.3. **Number-service**
- 3.4. **Register service**
- 3.5. **Legal service**
- 3.6. **Classification services**

[top of document](#)

3.4. Register service

The European Patent Register is the place where the EPO stores all the publicly available information on published European patent applications and international PCT applications designating the EPO as they pass through the grant procedure. OPS provides the interface for the [European Patent Register](#) online service. On this page, <http://www.epo.org/searching-for-patents/legal/register/documentation.html> you will find the technical documentation and an explanation of the tags.

The output of the register service is part of the `reg` namespace and defined in the [rplus.xsd](#) schema file.

Generic request structure:

```
GET http://ops.epo.org/rest-services/register/reference-type
e.g. publication OR application OR priority/input-format e.g.
epodoc/] input e.g. EP1000000 OR EP1000000.A1 OR ...

Accept: application/register+xml
```

3.4.1. Register retrieval

Request structure:

```
GET http://ops.epo.org/rest-services/register/reference-type:
publication OR application OR priority/input-format:epodoc/input
e.g. EP99203729

Accept: application/register+xml
```

Note, register service only supports the `epodoc` input format.

The **register retrieval** result can be influenced by using constituents to modify what you will get in your responses:

Constituent	Delivers
biblio	bibliographic data

procedural-steps	the data fields that have occurred during the granting procedure of the requested application; only relevant for register service
events	the information about actions that have taken place during the life of the patent application; only relevant for register service
upp	unitary patent protection

Table 24

Note, the default behaviour of the register retrieval is `biblio` so you don't have to add the `biblio` constituent if you want **to retrieve only bibliographic data**.

Register data without constituents (default: `biblio`)

Request:

<pre>GET http://ops.epo.org/rest-services/register/application/epodoc/EP99203729 Accept: application/register+xml</pre>
<pre>POST http://ops.epo.org/rest-services/register/application/epodoc/biblio Accept: application/register+xml Request Body: EP99203729</pre>

Response:

<pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?><?xml-stylesheet type='text/xsl' href='/style/rplus.xsl' ?> <ops:world-patent-data xmlns:ops="http://ops.epo.org" xmlns:reg="http://www.epo.org/register" xmlns:cpc="http://www.epo.org/cpcexport"> <ops:register-search total-result-count="1"> <ops:query syntax="CQL">publication=EP1000000</ops:query> <ops:range begin="1" end="1"/> <reg:register-documents produced-by="RO"> <reg:register-document status="NO OPPOSITION FILED WITHIN TIMELIMIT" produced-by="RO" lang="en" dtd-version="1.0" date-produced="20110510"> <reg:bibliographic-data status="NO OPPOSITION FILED WITHIN TIMELIMIT" lang="en" id="EP99203729P" country="EP"> <reg:publication-reference change-gazette-num="2000/20"> <reg:document-id lang="en"> <reg:country>EP</reg:country></pre>

```

        <reg:doc-number>1000000</reg:doc-number>
        <reg:kind>A1</reg:kind>
        <reg:date>20000517</reg:date>
    </reg:document-id>
</reg:publication-reference>
<reg:publication-reference change-gazette-num="2003/07">
    <reg:document-id lang="en">
        <reg:country>EP</reg:country>
        <reg:doc-number>1000000</reg:doc-number>
        <reg:kind>B1</reg:kind>
        <reg:date>20030212</reg:date>
    </reg:document-id>
</reg:publication-reference>
<reg:classifications-ipcr change-gazette-num="2000/20">
    <reg:classification-ipcr id="CLASSIFICATION_1">
        <reg:text>B28B5/02, B28B7/00, B28B1/29</reg:text>
    </reg:classification-ipcr>
</reg:classifications-ipcr>
<reg:application-reference change-gazette-num="2000/20">
    <reg:document-id>
        <reg:country>EP</reg:country>
        <reg:doc-number>99203729</reg:doc-number>
        <reg:date>19991108</reg:date>
    </reg:document-id>
</reg:application-reference>
<reg:language-of-filing>nl</reg:language-of-filing>
<reg:priority-claims change-gazette-num="2000/20">
    <reg:priority-claim sequence="1" kind="national">
        <reg:country>NL</reg:country>
        <reg:doc-number>19981010536</reg:doc-number>
        <reg:date>19981112</reg:date>
    </reg:priority-claim>
</reg:priority-claims>
<!-- Party elements removed for readability -->
<reg:designation-of-states change-gazette-num="2000/20">
    <reg:designation-pct>
        <reg:regional>
            <reg:region>
                <reg:country>EP</reg:country>
            </reg:region>
            <reg:country>AT</reg:country>
            <!-- Multiple countries removed for readability -->
        </reg:regional>
    </reg:designation-pct>
</reg:designation-of-states>
<reg:invention-title change-gazette-num="2000/20"
lang="en">Apparatus for manufacturing
green bricks for the brick manufacturing
industry</reg:invention-title>
<!-- Title elements in different languages removed-->
<reg:term-of-grant change-gazette-num="2008/17">
    <reg:lapsed-in-country>
        <reg:country>AT</reg:country>
        <reg:date>20030212</reg:date>
    </reg:lapsed-in-country>
    <!-- Multiple lapsed-in-countries removed -->
</reg:term-of-grant>

```

```
<!-- Multiple term-of-grant elements removed -->
<reg:dates-rights-effective>
  <reg:request-for-examination>
    <reg:date>20001027</reg:date>
  </reg:request-for-examination>
  <reg:first-examination-report-despatched>
    <reg:date>20020423</reg:date>
  </reg:first-examination-report-despatched>
</reg:dates-rights-effective>
<reg:references-cited>
  <reg:citation office="EP" cited-phase="search" id="cit_1">
    <reg:patcit
url="http://v3.espacenet.com/textdoc?DB=EPODOC&IDX=EP0680812&CY=EP">
    <reg:document-id>
      <reg:country>EP</reg:country>
      <reg:doc-number>0680812</reg:doc-number>
    </reg:document-id>
  </reg:patcit>
  <reg:category>A</reg:category>
</reg:citation>
  <reg:citation office="EP" cited-phase="search" id="cit_2">
    <reg:patcit
url="http://v3.espacenet.com/textdoc?DB=EPODOC&IDX=NL9400663&CY=NL">
    <reg:document-id>
      <reg:country>NL</reg:country>
      <reg:doc-number>9400663</reg:doc-number>
    </reg:document-id>
  </reg:patcit>
  <reg:category>A</reg:category>
</reg:citation>
  <reg:citation office="EP" cited-phase="search" id="cit_3">
    <reg:patcit
url="http://v3.espacenet.com/textdoc?DB=EPODOC&IDX=DE3546191&CY=DE">
    <reg:document-id>
      <reg:country>DE</reg:country>
      <reg:doc-number>3546191</reg:doc-number>
    </reg:document-id>
  </reg:patcit>
  <reg:category>A</reg:category>
</reg:citation>
</reg:references-cited>
<reg:opposition-data change-gazette-num="2004/06">
  <reg:opposition-not-filed>
    <reg:date>20031113</reg:date>
  </reg:opposition-not-filed>
</reg:opposition-data>
<reg:search-reports-information>
  <reg:search-report-information change-gazette-num="2000/20"
  declaration-of-no-search="not-determined" search-
type="national" office="EP">
    <reg:search-report-publication>
      <reg:document-id lang="en">
        <reg:country>EP</reg:country>
        <reg:doc-number>1000000</reg:doc-number>
```

```

        <reg:kind>A1</reg:kind>
        <reg:date>20000517</reg:date>
      </reg:document-id>
    </reg:search-report-publication>
  </reg:search-report-information>
</reg:search-reports-information>
</reg:bibliographic-data>
</reg:register-document>
</reg:register-documents>
</ops:register-search>
</ops:world-patent-data>

```

Register data with events

Request:

```
GET http://ops.epo.org/rest-services/register/application/epodoc/EP99203729/events
```

```
Accept: application/register+xml
```

```
POST http://ops.epo.org/rest-services/register/application/epodoc/events
```

```
Accept: application/register+xml
```

```
Request Body: EP99203729
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?><?xml-stylesheet
type='text/xsl' href='/style/rplus.xsl' ?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:reg="http://www.epo.org/register"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:register-search total-result-count="1">
    <ops:query syntax="CQL">application=EP99203729</ops:query>
    <ops:range begin="1" end="1"/>
    <reg:register-documents produced-by="RO">
      <reg:register-document status="NO OPPOSITION FILED WITHIN TIMELIMIT"
produced-by="RO"
      lang="en" dtd-version="1.0" date-produced="20110510">
        <reg:events-data>
          <reg:dossier-event event-type="new" id="EVT_28">
            <reg:event-date>
              <reg:date>20000331</reg:date>
            </reg:event-date>
            <reg:event-code>0009012</reg:event-code>
            <reg:event-text event-text-type="DESCRIPTION">Publication in
section I.1 EP
            Bulletin</reg:event-text>
          </reg:dossier-event>
        </reg:events-data>
      </reg:register-document>
    </reg:register-documents>
  </ops:register-search>
</ops:world-patent-data>

```

```

        <reg:gazette-reference>
          <reg:gazette-num>2000/20</reg:gazette-num>
          <reg:date>20000517</reg:date>
        </reg:gazette-reference>
      </reg:dossier-event>
    </reg:events-data>
  <reg:events-data>
    <reg:dossier-event event-type="new" id="EVT_29">
      <reg:event-date>
        <reg:date>20000526</reg:date>
      </reg:event-date>
      <reg:event-code>0009199FREP</reg:event-code>
      <reg:event-text event-text-type="DESCRIPTION">Change -
representative</reg:event-text>
      <reg:gazette-reference>
        <reg:gazette-num>2000/28</reg:gazette-num>
        <reg:date>20000712</reg:date>
      </reg:gazette-reference>
    </reg:dossier-event>
  </reg:events-data>
  <!-- Event 3-27 removed for readability -->
</reg:register-document>
</reg:register-documents>
</ops:register-search>
</ops:world-patent-data>

```

Register data with procedural-steps

Request:

```
GET http://ops.epo.org/rest-
services/register/application/epodoc/EP99203729/procedural-steps
```

```
Accept: application/register+xml
```

```
POST http://ops.epo.org/rest-
services/register/application/epodoc/procedural-steps
```

```
Accept: application/register+xml
```

```
Request Body: EP99203729
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?><?xml-stylesheet
type='text/xsl' href='/style/rplus.xsl' ?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:reg="http://www.epo.org/register"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:register-search total-result-count="1">
    <ops:query syntax="CQL">application=EP99203729</ops:query>

```

```

<ops:range begin="1" end="1"/>
<reg:register-documents produced-by="RO">
  <reg:register-document status="NO OPPOSITION FILED WITHIN TIMELIMIT"
produced-by="RO"
  lang="en" dtd-version="1.0" date-produced="20110510">
    <reg:procedural-data>
      <reg:procedural-step procedure-step-phase="undefined"
id="RENEWAL_5">
        <reg:procedural-step-code>RFEE</reg:procedural-step-code>
        <reg:procedural-step-text step-text-
type="STEP_DESCRIPTION">Renewal fee
          payment</reg:procedural-step-text>
        <reg:procedural-step-text step-text-
type="YEAR">03</reg:procedural-step-text>
        <reg:procedural-step-date step-date-type="DATE_OF_PAYMENT">
          <reg:date>20011128</reg:date>
        </reg:procedural-step-date>
      </reg:procedural-step>
      <reg:procedural-step procedure-step-phase="undefined"
id="RENEWAL_6">
        <reg:procedural-step-code>RFEE</reg:procedural-step-code>
        <reg:procedural-step-text step-text-
type="STEP_DESCRIPTION">Renewal fee
          payment</reg:procedural-step-text>
        <reg:procedural-step-text step-text-
type="YEAR">04</reg:procedural-step-text>
        <reg:procedural-step-date step-date-type="DATE_OF_PAYMENT">
          <reg:date>20021128</reg:date>
        </reg:procedural-step-date>
      </reg:procedural-step>
      <reg:procedural-step procedure-step-phase="examination"
id="STEP_IGRA_3">
        <reg:procedural-step-code>IGRA</reg:procedural-step-code>
        <reg:procedural-step-text step-text-
type="STEP_DESCRIPTION">Intention to grant the
          patent</reg:procedural-step-text>
        <reg:procedural-step-date step-date-type="DATE_OF_DISPATCH">
          <reg:date>20020807</reg:date>
        </reg:procedural-step-date>
        <reg:procedural-step-date step-date-type="GRANT_FEE_PAID">
          <reg:date>20020824</reg:date>
        </reg:procedural-step-date>
        <reg:procedural-step-date step-date-type="PRINT_FEE_PAID">
          <reg:date>20020824</reg:date>
        </reg:procedural-step-date>
      </reg:procedural-step>
      <reg:procedural-step procedure-step-phase="examination"
id="STEP_AGRA_3">
        <reg:procedural-step-code>AGRA</reg:procedural-step-code>
        <reg:procedural-step-text step-text-
type="STEP_DESCRIPTION">Announcement of grant (old
          EPC before 1 July 2002)</reg:procedural-step-text>
        <reg:procedural-step-text step-text-
type="approval">yes</reg:procedural-step-text>
        <reg:procedural-step-date step-date-type="DATE_OF_DISPATCH">
          <reg:date>20020423</reg:date>
        </reg:procedural-step-date>

```

```

        </reg:procedural-step>
        <reg:procedural-step procedure-step-phase="examination"
id="STEP_PROL_3">
        <reg:procedural-step-code>PROL</reg:procedural-step-code>
        <reg:procedural-step-text step-text-
type="STEP_DESCRIPTION">Language of the
        procedure</reg:procedural-step-text>
        <reg:procedural-step-text step-text-type="procedure language"
        >en</reg:procedural-step-text>
        </reg:procedural-step>
    </reg:procedural-data>
</reg:register-document>
</reg:register-documents>
</ops:register-search>
</ops:world-patent-data>

```

Register data with upp

Request:

```

GET http://ops.epo.org/rest-
services/register/publication/epodoc/EP99203729/upp
Accept: application/register+xml

```

```

POST http://ops.epo.org/rest-
services/register/application/epodoc/upp
Accept: application/register+xml
Request Body: EP99203729

```

Response, truncated for UPP data only:

```

<!--Biblio information truncated -->
<reg:unitary-patent>
  <reg:unitary-patent-statuses>
    <reg:unitary-patent-status change-date="20160309" status-code="6">Request
for unitary effect filed</reg:unitary-patent-status>
  </reg:unitary-patent-statuses>
  <reg:procedural-data>
    <reg:procedural-step id="STEP_UP_UDFI_1" procedure-step-phase="unitary-
protection">
      <reg:procedural-step-code>UDFI</reg:procedural-step-code>

```

```

        <reg:procedural-step-text step-text-type="STEP_DESCRIPTION">The
request for unitary effects</reg:procedural-step-text>
        <reg:procedural-step-date step-date-type="DATE_OF_REQUEST">
            <reg:date>20150415</reg:date>
        </reg:procedural-step-date>
        <reg:gazette-reference>
            <reg:gazette-num>2016/15</reg:gazette-num>
            <reg:date>
</reg:date>
                </reg:gazette-reference>
            </reg:procedural-step>
        </reg:procedural-data>
        <reg:events-data>
            <reg:dossier-event id="EVT_UP_2" event-type="new">
                <reg:event-date>
                    <reg:date>20160309</reg:date>
                </reg:event-date>
                <reg:event-code>0009700UREQ01</reg:event-code>
                <reg:event-text event-text-type="DESCRIPTION">Filing of request for
unitary effect</reg:event-text>
                <reg:gazette-reference>
                    <reg:gazette-num>2016/15</reg:gazette-num>
                    <reg:date>20160413</reg:date>
                </reg:gazette-reference>
            </reg:dossier-event>
        </reg:events-data>
    </reg:unitary-patent>

```

List of all events that European Patent Register and consequently the OPS Register will present can be found here:

<https://register.epo.org/allEvents?lng=en>

See official notification on EP Register and UPP/UPC related information here: <https://www.epo.org/searching-for-patents/legal/register/archive.html>

Register data with biblio, events and procedural-steps

Request:

```
GET http://ops.epo.org/rest-services/register/application/epodoc/EP99203729/biblio,events,procedural-steps
```

```
Accept: application/register+xml
```

```
POST http://ops.epo.org/rest-services/register/application/epodoc/biblio,events,procedural-steps
```

```
Accept: application/register+xml
```

```
Request Body: EP99203729
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?><?xml-stylesheet
type='text/xsl' href='/style/rplus.xsl' ?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:reg="http://www.epo.org/register"
  xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:register-search total-result-count="1">
    <ops:query syntax="CQL">application=EP99203729</ops:query>
    <ops:range begin="1" end="1"/>
    <reg:register-documents produced-by="RO">
      <reg:register-document status="NO OPPOSITION FILED WITHIN TIMELIMIT"
produced-by="RO"
        lang="en" dtd-version="1.0" date-produced="20110510">
        <reg:bibliographic-data status="NO OPPOSITION FILED WITHIN
TIMELIMIT" lang="en"
          id="EP99203729P" country="EP">
          <reg:publication-reference change-gazette-num="2000/20">
            <reg:document-id lang="en">
              <reg:country>EP</reg:country>
              <reg:doc-number>1000000</reg:doc-number>
              <reg:kind>A1</reg:kind>
              <reg:date>20000517</reg:date>
            </reg:document-id>
          </reg:publication-reference>
          <reg:publication-reference change-gazette-num="2003/07">
            <reg:document-id lang="en">
              <reg:country>EP</reg:country>
              <reg:doc-number>1000000</reg:doc-number>
              <reg:kind>B1</reg:kind>
              <reg:date>20030212</reg:date>
            </reg:document-id>
          </reg:publication-reference>
        <!-- remaining bibliographic data removed for readability -->
```



```

</reg:bibliographic-data>
<reg:procedural-data>
  <reg:procedural-step procedure-step-phase="undefined"
id="RENEWAL_7">
    <reg:procedural-step-code>RFEE</reg:procedural-step-code>
    <reg:procedural-step-text step-text-
type="STEP_DESCRIPTION">Renewal fee
      payment</reg:procedural-step-text>
    <reg:procedural-step-text step-text-
type="YEAR">03</reg:procedural-step-text>
    <reg:procedural-step-date step-date-type="DATE_OF_PAYMENT">
      <reg:date>20011128</reg:date>
    </reg:procedural-step-date>
    </reg:procedural-step>
    <!-- remaining procedural steps removed for readability -->
  </reg:procedural-data>
  <reg:events-data>
    <reg:dossier-event event-type="new" id="EVT_82">
      <reg:event-date>
        <reg:date>20000331</reg:date>
      </reg:event-date>
      <reg:event-code>0009012</reg:event-code>
      <reg:event-text event-text-type="DESCRIPTION">Publication in
section I.1 EP
      Bulletin</reg:event-text>
      <reg:gazette-reference>
        <reg:gazette-num>2000/20</reg:gazette-num>
        <reg:date>20000517</reg:date>
      </reg:gazette-reference>
    </reg:dossier-event>
  </reg:events-data>
  <!-- remaining events removed for readability -->
</reg:register-document>
</reg:register-documents>
</ops:register-search>
</ops:world-patent-data>

```

3.4.2. Register search

Use this service to find specific register data -- that is part of the public aspect of the patent lifecycle.

The OPS register service uses the same search operators, input/output formats and CQL identifiers as the European Patent Register online service. It is advisable to have a look at the [search query information](#) to understand how to construct a query. **Note**, these CQL identifiers are for the Register service only. **Do not use the CQL details for Published-data service.**

EP Register with UPP constituent offers more search identifiers and they are listed here: <https://register.epo.org/help?lng=en&topic=UPsmartsearch>.

Request structure:

```
GET http://ops.epo.org/rest-
services/register/search{constituents}/?q=[QUERY STRING]
```

```
Accept: application/register+xml
```

Request example:

```
GET http://ops.epo.org/rest-services/register/search/?q=pa%3DIBM
```

```
Accept: application/register+xml
```

```
POST http://ops.epo.org/rest-services/register/search
```

```
Accept: application/register+xml
```

```
Request Body: q=pa%3DIBM
```

Response:

Note, that the response has 763 applications containing “International Business Machines” as applicant, contains a shorter version of biblio data than retrieval:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?><?xml-stylesheet
type='text/xsl' href='/style/rplus.xsl' ?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:reg="http://www.epo.org/register"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:register-search total-result-count="781">
    <ops:query syntax="CQL">pa=IBM</ops:query>
    <ops:range begin="1" end="25"/>
    <reg:register-documents produced-by="RO">
      <reg:register-document>
        <reg:bibliographic-data>
          <reg:publication-reference>
            <reg:document-id>
              <reg:country>WO</reg:country>
              <reg:doc-number>2011042229</reg:doc-number>
              <reg:date>20110414</reg:date>
            </reg:document-id>
          </reg:publication-reference>
          <reg:classifications-ipcr>
            <reg:classification-ipcr sequence="CLASSIFICATION_1">
              <reg:text>H04N5/235</reg:text>
            </reg:classification-ipcr>
          </reg:classifications-ipcr>
          <reg:application-reference>
            <reg:document-id>
              <reg:country>EP</reg:country>
```

```
<reg:doc-number>10730472</reg:doc-number>
</reg:document-id>
</reg:application-reference>
<reg:parties>
  <reg:applicants>
    <reg:applicant app-type="applicant">
      <reg:addressbook>
        <reg:name>International Business Machines
Corporation</reg:name>
      </reg:addressbook>
    </reg:applicant>
  </reg:applicants>
</reg:parties>
</reg:bibliographic-data>
</reg:register-document>
<!-- 24 more applications -->
</reg:register-documents>
</ops:register-search>
</ops:world-patent-data>
```

Range control

The default range of the register search response is 1-25. It can be changed by specifying an HTTP header parameter **Range=[begin_range-end_range]**.

For test purposes, it is also possible to use the range query parameter by adding **&Range=[begin_range-end_range]** to the end of your URI (e.g. <http://ops.epo.org/rest-services/register/search/?q=pa%3DIBM&Range=1-10>).

Request Example

```
GET http://ops.epo.org/rest-services/register/search/?q=pa%3DIBM
```

```
Accept: application/register+xml
```

```
Range: 1-10
```

```
<ops:register-search total-result-count="781">  
  <ops:query syntax="CQL">pa=IBM</ops:query>  
  <ops:range begin="1" end="10"/>  
</ops:register-search>
```

Note, the maximum range is 100.

- 3.1. **Published-data services**
- 3.2. **Family service**
- 3.3. **Number-service**
- 3.4. **Register service**
- 3.5. **Legal service**
- 3.6. **Classification services**

[top of document](#)

3.5. Legal service

This service retrieves the legal data of the patent lifecycle, including the register domain of the patent.

Generic Request structure:

```
GET http://ops.epo.org/rest-services/  
legal/reference-type/input-type  
  
Accept: application/ops+xml
```

Note, bold means mandatory.

Response structure:

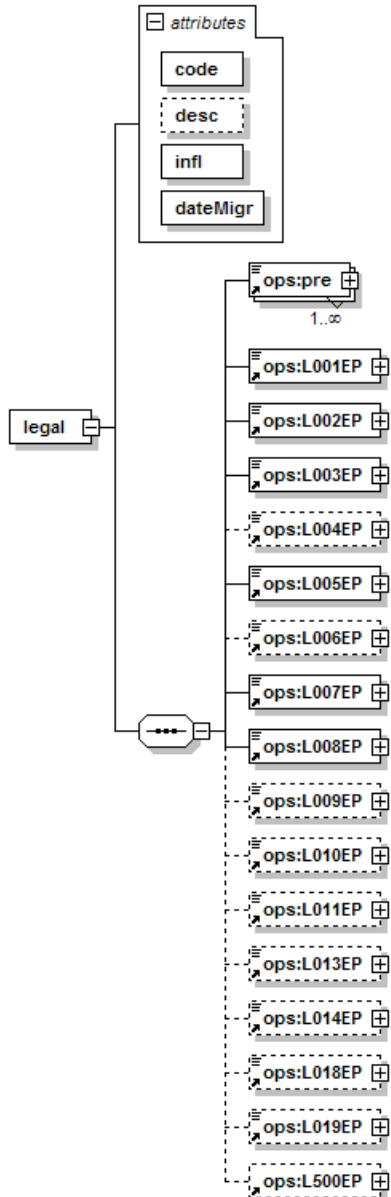


Figure 33

The structure shown in Figure 33 is part of the `ops` namespace and defined in the [ops_legal.xsd](#) schema file.

Request example:

```
GET http://ops.epo.org/rest-services/legal/publication/docdb/EP.1000000.A1
Accept: application/ops+xml
```

POST <http://ops.epo.org/rest-services/legal/publication/docdb>

Accept: application/ops+xml

Request Body: EP.1000000.A1

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns="http://www.epo.org/exchange">
  <ops:patent-family legal="true">
    <ops:publication-reference>
      <document-id document-id-type="docdb">
        <country>EP</country>
        <doc-number>1000000</doc-number>
        <kind>A1</kind>
      </document-id>
    </ops:publication-reference>
    <ops:family-member family-id="19768124">
      <publication-reference>
        <document-id document-id-type="docdb">
          <country>EP</country>
          <doc-number>1000000</doc-number>
          <kind>A1</kind>
          <date>20000517</date>
        </document-id>
        <document-id document-id-type="epodoc">
          <doc-number>EP1000000</doc-number>
          <date>20000517</date>
        </document-id>
      </publication-reference>
      <application-reference doc-id="17397285" is-representative="YES">
        <document-id document-id-type="docdb">
          <country>EP</country>
          <doc-number>99203729</doc-number>
          <kind>A</kind>
          <date>19991108</date>
        </document-id>
      </application-reference>
      <priority-claim kind="national">
        <document-id document-id-type="docdb">
          <country>NL</country>
          <doc-number>1010536</doc-number>
          <kind>A</kind>
          <date>19981112</date>
        </document-id>
      </priority-claim>
      <priority-active-indicator>YES</priority-active-indicator>
    </ops:legal code="AK" desc="DESIGNATED CONTRACTING STATES:" infl="+
dateMigr="00010101">
      <ops:pre line="00001">EP 99203729A 2000-05-17AK +DESIGNATED
CONTRACTING STATES: Kind Code of Ref Document A1</ops:pre>
```

```

<ops:pre line="00002">EP      99203729A 2000-05-17AK      +DESIGNATED
CONTRACTING STATES: Designated State(s) AT BE CH CY DE DK ES FI FR GB GR IE
IT LI LU MC NL PT SE</ops:pre>
  <ops:L001EP desc="Country Code">EP</ops:L001EP>
  <ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
  <ops:L003EP desc="Document Number"> 99203729</ops:L003EP>
  <ops:L004EP desc="Kind Code">A</ops:L004EP>
  <ops:L005EP desc="IPR Type">PI</ops:L005EP>
  <ops:L006EP desc="PRS DOCUMENT TYPE">P</ops:L006EP>
  <ops:L007EP desc="Gazette DATE">2000-05-17</ops:L007EP>
  <ops:L008EP desc="Legal Event Code 1">AK</ops:L008EP>
  <ops:L018EP desc="DATE last exchanged">2003-01-01</ops:L018EP>
  <ops:L019EP desc="DATE first created">2003-05-28</ops:L019EP>
  <ops:L500EP>
    <ops:L506EP desc="Kind Code of Ref Document">A1</ops:L506EP>
    <ops:L507EP desc="Designated State(s)">AT BE CH CY DE DK ES FI FR GB
GR IE IT LI LU MC NL PT SE</ops:L507EP>
  </ops:L500EP>
</ops:legal>
  <ops:legal code="AX      " desc="EXTENSION OF THE EUROPEAN PATENT TO"
infl="+      dateMigr="00010101">
  <ops:pre line="00001">EP      99203729A 2000-05-17AX      +EXTENSION OF
THE EUROPEAN PATENT TO Free Format Text AL;LT;LV;MK;RO;SI</ops:pre>
  <ops:L001EP desc="Country Code">EP</ops:L001EP>
  <ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
  <ops:L003EP desc="Document Number"> 99203729</ops:L003EP>
  <ops:L004EP desc="Kind Code">A</ops:L004EP>
  <ops:L005EP desc="IPR Type">PI</ops:L005EP>
  <ops:L006EP desc="PRS DOCUMENT TYPE">P</ops:L006EP>
  <ops:L007EP desc="Gazette DATE">2000-05-17</ops:L007EP>
  <ops:L008EP desc="Legal Event Code 1">AX</ops:L008EP>
  <ops:L018EP desc="DATE last exchanged">2003-01-01</ops:L018EP>
  <ops:L019EP desc="DATE first created">2003-05-28</ops:L019EP>
  <ops:L500EP>
    <ops:L510EP desc="Free Format Text">AL;LT;LV;MK;RO;SI</ops:L510EP>
  </ops:L500EP>
</ops:legal>
  <!-- other legal elements removed for readability -->
</ops:family-member>
</ops:patent-family>
</ops:world-patent-data>

```


- 3.1. **Published-data services**
- 3.2. **Family service**
- 3.3. **Number-service**
- 3.4. **Register service**
- 3.5. **Legal service**
- 3.6. **Classification services**

[top of document](#)

3.6. Classification services

Since January 2013 the European Patent Office, in partnership with the US patent office, classifies and delivers data using the Cooperative Patent Classification.

The 3.0 implementation of the OPS service retrieves the Cooperative classification (CPC) scheme data.

CPC is divided into eight sections (A-H) that are sub-divided into classes, sub-classes, groups and sub-groups. As part of the official CPC, OPS allows to access the so-called Y indexes. These are subclasses, which relate to specific clean energy technologies, namely Y02C (greenhouse gases- capture and storage/sequestration or disposal) and Y02E (greenhouse gases - emissions reduction technologies related to energy generation, transmission or distribution). You can find more information to these indexes here: <http://www.epo.org/news-issues/issues/classification/classification.html>.

Generic Request structure:

GET http://ops.epo.org/rest-services/classification/cpc/[classification-symbol]?query-string
Accept: application/xml
POST http://ops.epo.org/rest-services/classification/cpc/
Accept: application/cpc+xml
Request Body: A01B

Note, bold means mandatory.

Response structure:

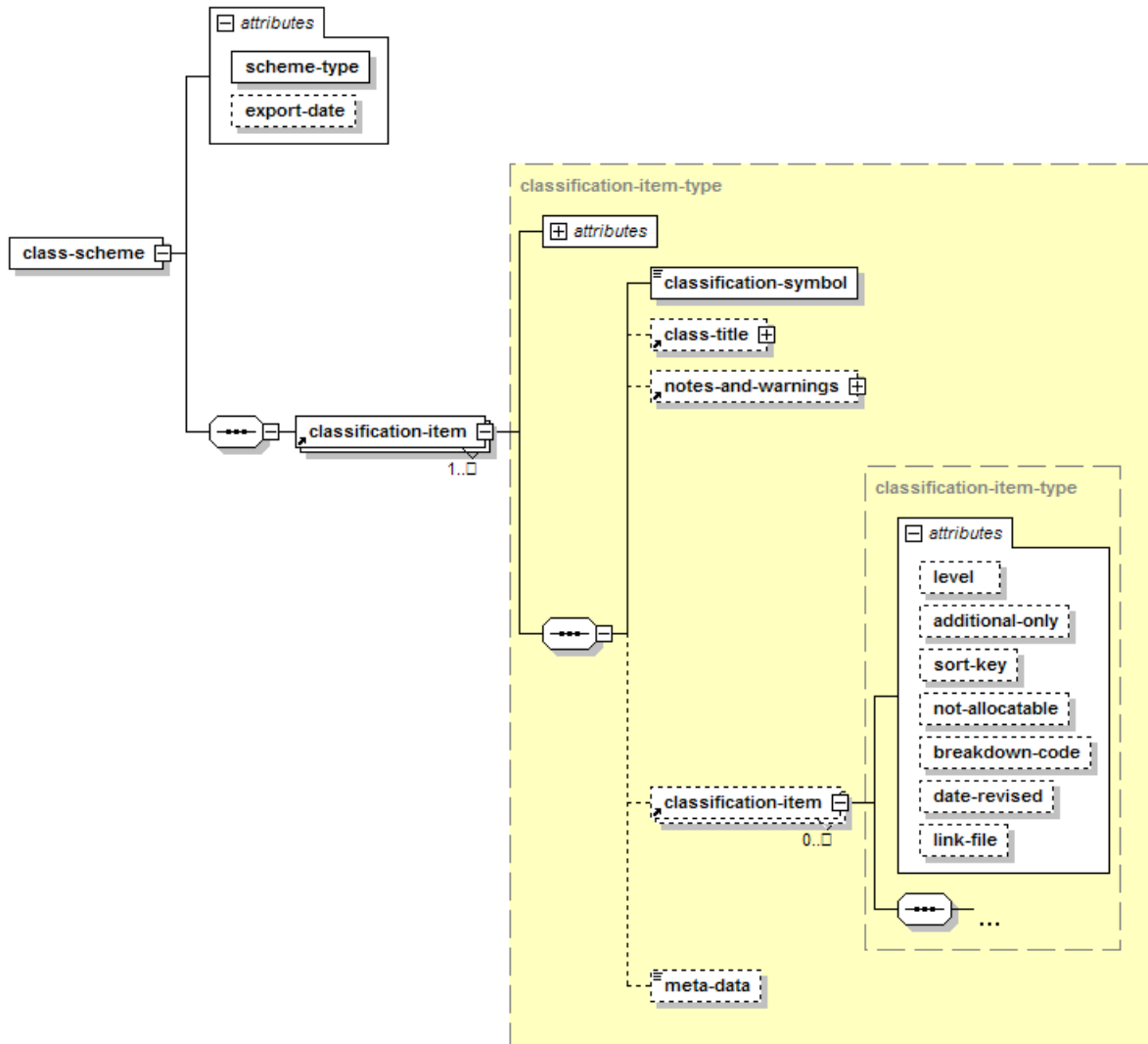


Figure 34

The structure shown Figure 34 is part of the cpc namespace and defined in the [CPCSchema.xsd](#) schema file.

The classification response is `ops:classification-scheme`.

3.6.1.CPC Retrieval

Note, for more general information on the Cooperative Classification, please refer to: <http://www.cooperativepatentclassification.org>

Valid Query-string parameters in the CPC service

Query-string	Description
--------------	-------------

depth	Determines how many children elements should be included in the response
ancestors	Includes symbols above the requested element
navigation	Includes navigation symbols next & previous in the response

Table 25

Retrieval by specifying the complete classification symbol:

```
GET http://ops.epo.org/rest-services/classification/cpc/\[classification-symbol\]
```

```
Accept: application/cpc+xml
```

Retrieval of a specific number of child levels

The depth parameter is used to specify how many levels of children of the requested symbol should be included into the response.

Note, requesting "depth" > 1 (more than one) or "all" is allowed only if the requested symbol belongs to a level > 5.

```
GET http://ops.epo.org/rest-services/classification/cpc/\[classification-symbol\]?depth=x
```

```
Accept: application/cpc+xml
```

Note, x = numeric value or "all" for the retrieval of all children of the given classification.

Below are examples for retrieving classifications.

Request for classification A:

```
GET http://ops.epo.org/rest-services/classification/cpc/A
```

```
Accept: application/cpc+xml
```

```
POST http://ops.epo.org/rest-services/classification/cpc
```

```
Accept: application/cpc+xml
```

```
Request Body: A
```

Response (which returns the top level of classification A, 'Human Necessities'):

```

<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:cpc>
      <cpc:class-scheme scheme-type="cpc" export-date="2012-10-13">
        <cpc:classification-item level="2" additional-only="false"
sort-key="A" not-allocatable="false" breakdown-code="false" date-
revised="2012-10-12" link-file="classification/cpc/A">
          <cpc:classification-symbol>A</cpc:classification-symbol>
          <cpc:class-title date-revised="2012-10-12">
            <cpc:title-part>
              <cpc:text scheme="ipc">HUMAN
NECESSITIES</cpc:text>
            </cpc:title-part>
          </cpc:class-title>
        </cpc:classification-item>
      </cpc:class-scheme>
    </ops:cpc>
  </ops:classification-scheme>
</ops:world-patent-data>

```

Request for the tenth level of the A classification which signifies 'Thermally sensitive initiators':

```
GET http://ops.epo.org/rest-services/classification/cpc/A62C37/48
```

```
Accept: application/cpc+xml
```

```
POST http://ops.epo.org/rest-services/classification/cpc
```

```
Accept: application/cpc+xml
```

```
Request Body: A62C37/48
```

Response example

```

<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:cpc>
      <cpc:class-scheme scheme-type="cpc" export-date="2012-10-13">
        <cpc:classification-item level="10" additional-only="false"
sort-key="A62C37/48" not-allocatable="false" breakdown-code="false" date-
revised="2012-10-12" link-file="classification/cpc/A62C37/48">
          <cpc:classification-
symbol>A62C37/48</cpc:classification-symbol>
          <cpc:class-title date-revised="2012-10-12">

```

```

                <cpc:title-part>
                <cpc:text scheme="ipc">Thermally sensitive
initiators</cpc:text>
                </cpc:title-part>
            </cpc:class-title>
        </cpc:classification-item>
    </cpc:class-scheme>
</ops:cpc>
</ops:classification-scheme>
</ops:world-patent-data>

```

Request for the next and previous entries for classification A01:

```

GET http://ops.epo.org/rest-services/classification/cpc/A01?navigation
Accept: application/cpc+xml

```

Response:

```

<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:cpc>
      <ops:navigation>
        <ops:prev xlink:href="classification/cpc/A">A</ops:prev>
        <ops:next xlink:href="classification/cpc/A01B">A01B</ops:next>
      </ops:navigation>
      <cpc:class-scheme scheme-type="cpc" export-date="2012-10-13">
        <cpc:classification-item level="4" additional-only="false" sort-
key="A01" not-allocatable="false" breakdown-code="false" date-revised="2012-
10-12" link-file="classification/cpc/A01">
          <cpc:classification-symbol>A01</cpc:classification-symbol>
          <cpc:class-title date-revised="2012-10-12">
            <cpc:title-part>
              <cpc:text scheme="ipc">AGRICULTURE</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> FORESTRY</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> ANIMAL HUSBANDRY</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> HUNTING</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> TRAPPING</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> FISHING</cpc:text>
            </cpc:title-part>
          </cpc:class-title>
        </cpc:classification-item>
      </cpc:class-scheme>

```

```

</ops:cpc>
</ops:classification-scheme>
</ops:world-patent-data>

```

Request for the ancestors and navigation items classification A01:

```

GET http://ops.epo.org/rest-
services/classification/cpc/A01?navigation&ancestors

Accept: application/cpc+xml

```

Response:

```

<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:cpc>
      <ops:navigation>
        <ops:prev xlink:href="classification/cpc/A">A</ops:prev>
        <ops:next xlink:href="classification/cpc/A01B">A01B</ops:next>
      </ops:navigation>
      <cpc:class-scheme scheme-type="cpc" export-date="2012-10-13">
        <cpc:classification-item level="4" additional-only="false" sort-
key="A01" not-allocatable="false" breakdown-code="false" date-revised="2012-
10-12" link-file="classification/cpc/A01">
          <cpc:classification-symbol>A01</cpc:classification-symbol>
          <cpc:class-title date-revised="2012-10-12">
            <cpc:title-part>
              <cpc:text scheme="ipc">AGRICULTURE</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> FORESTRY</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> ANIMAL HUSBANDRY</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> HUNTING</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> TRAPPING</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> FISHING</cpc:text>
            </cpc:title-part>
          </cpc:class-title>
        </cpc:classification-item>
      </cpc:class-scheme>
    </ops:cpc>
  </ops:classification-scheme>
</ops:world-patent-data>

```

Request for all children one level below classification A:

```
GET http://ops.epo.org/rest-services/classification/cpc/A?depth=1
```

```
Accept: application/cpc+xml
```

```
POST http://ops.epo.org/rest-services/classification/cpc/?depth=1
```

```
Accept: application/cpc+xml
```

```
Request Body: A
```

Response:

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:cpc>
      <cpc:class-scheme scheme-type="cpc" export-date="2012-10-13">
        <cpc:classification-item level="2" additional-only="false" sort-
key="A" not-allocatable="false" breakdown-code="false" date-revised="2012-
10-12" link-file="classification/cpc/A">
          <cpc:classification-symbol>A</cpc:classification-symbol>
          <cpc:class-title date-revised="2012-10-12">
            <cpc:title-part>
              <cpc:text scheme="ipc">HUMAN NECESSITIES</cpc:text>
            </cpc:title-part>
          </cpc:class-title>
          <cpc:classification-item level="3" additional-only="false" sort-
key="A01" not-allocatable="false" breakdown-code="false" date-revised="2012-
10-12" link-file="classification/cpc/A01">
            <cpc:classification-symbol>A01</cpc:classification-symbol>
            <cpc:class-title date-revised="2012-10-12">
              <cpc:title-part>
                <cpc:text scheme="ipc">Agriculture</cpc:text>
              </cpc:title-part>
            </cpc:class-title>
            <cpc:classification-item level="4" additional-only="false" sort-
key="A01" not-allocatable="false" breakdown-code="false" date-revised="2012-
10-12" link-file="classification/cpc/A01">
              <cpc:classification-symbol>A01</cpc:classification-symbol>
              <cpc:class-title date-revised="2012-10-12">
                <cpc:title-part>
                  <cpc:text scheme="ipc">AGRICULTURE</cpc:text>
                </cpc:title-part>
                <cpc:title-part>
                  <cpc:text scheme="ipc"> FORESTRY</cpc:text>
                </cpc:title-part>
                <cpc:title-part>
                  <cpc:text scheme="ipc"> ANIMAL HUSBANDRY</cpc:text>
                </cpc:title-part>
              </cpc:class-title>
            </cpc:classification-item>
          </cpc:classification-item>
        </cpc:classification-item>
      </cpc:class-scheme>
    </ops:cpc>
  </ops:classification-scheme>
</ops:world-patent-data>
```

```

        <cpc:title-part>
          <cpc:text scheme="ipc"> HUNTING</cpc:text>
        </cpc:title-part>
        <cpc:title-part>
          <cpc:text scheme="ipc"> TRAPPING</cpc:text>
        </cpc:title-part>
        <cpc:title-part>
          <cpc:text scheme="ipc"> FISHING</cpc:text>
        </cpc:title-part>
      </cpc:class-title>
    <cpc:meta-data>+</cpc:meta-data>
  </cpc:classification-item>
  <cpc:meta-data>+</cpc:meta-data>
</cpc:classification-item>
</cpc:classification-item>
<!-- 3 more classification-item elements removed for readability of the
example -->
  </cpc:class-scheme>
</ops:cpc>
</ops:classification-scheme>
</ops:world-patent-data>

```

3.6.2.CPC Media retrieval

With the OPS classification service it is possible to retrieve CPC media referenced in the classification text in the format specified.

Request structure:

```
GET http://ops.epo.org/rest-services/classification/cpc/media/\[image-name\]
```

```
Accept: [see list of accepted media formats below]
```

The media name and type can be extracted from the CPC retrieval response.

List of accepted media formats:

Format	Media type
GIF	image/gif
JPEG	image/jpeg
TIF	image/tif
BMP	image/x-ms-bmp
PNG	image/png
WAV	audio/wav
WMA	audio/x-ms-wma
WMV	video/x-ms-wmv
MP3	audio/mpeg

Format	Media type
UNKNOWN	type/x-unknown

Table 26

Example

Use the classification retrieval service, extract the media name and type from the response (e.g. **A01N37/12** symbol):

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:cpc>
      <cpc:class-scheme scheme-type="cpc" export-date="2012-10-13">
        <cpc:classification-item level="8" additional-only="false" sort-
key="A01N37/12" not-allocatable="false" breakdown-code="false" date-
revised="2012-10-12" link-file="classification/cpc/A01N37/12">
          <cpc:classification-symbol>A01N37/12</cpc:classification-symbol>
          <cpc:class-title date-revised="2012-10-12">
            <cpc:title-part>
              <cpc:text scheme="ipc">containing the group <cpc:media
id="classification/cpc/media/100.gif" type="gif"/>, wherein Cn means a
carbon skeleton not containing a ring</cpc:text>
            </cpc:title-part>
            <cpc:title-part>
              <cpc:text scheme="ipc"> Thio analogues thereof</cpc:text>
            </cpc:title-part>
          </cpc:class-title>
        </cpc:classification-item>
      </cpc:class-scheme>
    </ops:cpc>
  </ops:classification-scheme>
</ops:world-patent-data>
```

Request example:

```
GET http://ops.epo.org/rest-services/classification/cpc/media/1000.gif
Accept: image/gif
```

Response: the image in the GIF format will be in response

3.6.3.CPC Search

Search for keywords in the Cooperative classifications.

If you don't know the name of the CPC symbol you are looking for, you can use the CPC search service to identify possible interesting CPC symbols by searching for keywords in the title and abstract fields of the Espacenet database.

The result will be a list of CPC classes, each showing a percentage value. The higher the percentage value, the most likely it is that the CPC class corresponds to the concept searched.

By default, only the first 10 CPC classes with the highest percentage value are shown.

The CPC search basically uses the same [CQL syntax](#) as the biblio search but is restricted to titles and abstracts only.

Request structure:

```
GET http://ops.epo.org/rest-
services/classification/cpc/search/?q=[QUERY STRING]
```

```
Accept: application/cpc+xml
```

Response structure:

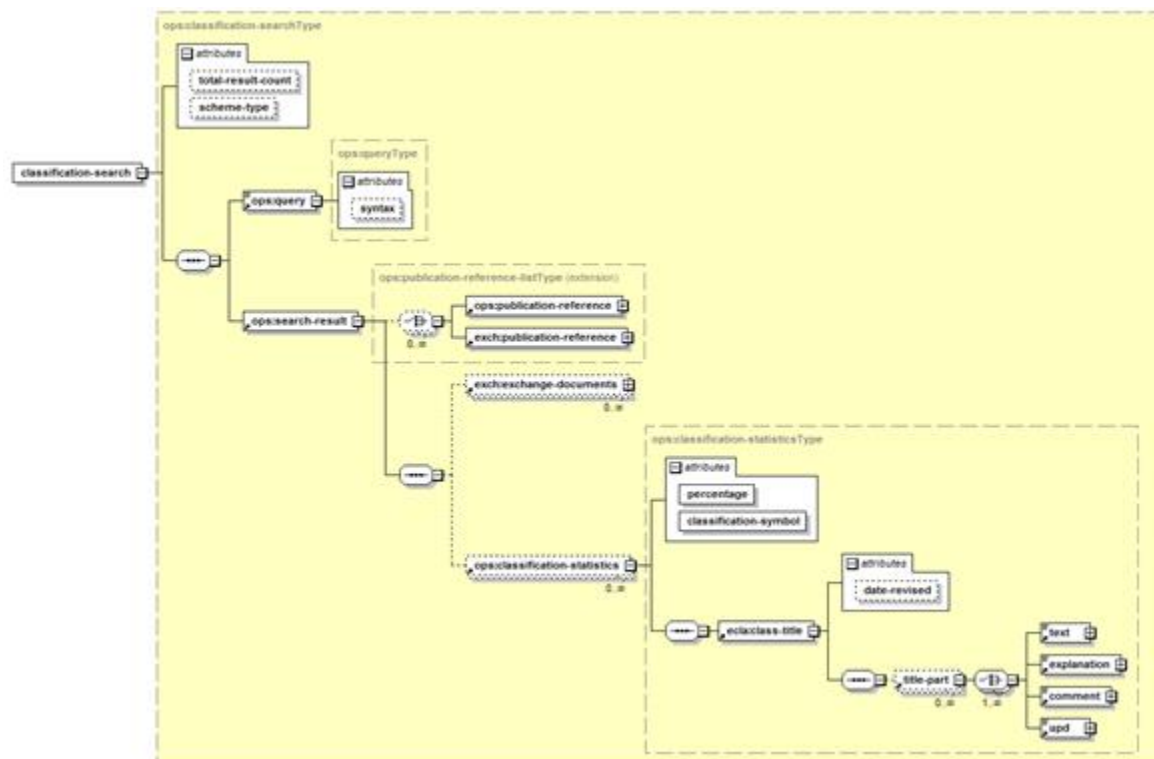


Figure 35

The structure shown in Figure 35 is part of the `ops` namespace and defined in the [ops.xsd schema file](#).

Request example:

```
GET http://ops.epo.org/rest-services/classification/cpc/search?q=chemistry
```

Accept: application/cpc+xml

POST <http://ops.epo.org/rest-services/classification/cpc/search>

Accept: application/ops+xml

Request Body: **chemistry**

Note, as the CPC search is restricted to the title and abstract, you don't have to provide the index name (titleandabstract) in your query. However, `q=titleandabstract%3Dchemistry` can also be used and results in the same response.

Response:

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:reg="http://www.epo.org/register"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-search total-result-count="10" scheme-type="cpc">
    <ops:query syntax="CQL">titleandabstract = chemistry</ops:query>
    <ops:search-result>
      <ops:classification-statistics classification-symbol="A61B10/00"
percentage="2.8985507">
        <cpc:class-title date-revised="2012-10-12">
          <cpc:title-part>
            <cpc:text scheme="ipc">
Other methods or instruments for diagnosis, e.g. instruments for taking a
cell sample, for biopsy, for vaccination diagnosis
</cpc:text>
              <cpc:explanation>
                <cpc:text scheme="ipc">
vaccination prophylaxis, vaccination therapy
<cpc:class-ref scheme="cpc">A61B17/20</cpc:class-ref>
              </cpc:text>
            </cpc:explanation>
          </cpc:title-part>
          <cpc:title-part>
            <cpc:text scheme="ipc">Sex determination</cpc:text>
          </cpc:title-part>
          <cpc:title-part>
            <cpc:text scheme="ipc">Ovulation-period determination</cpc:text>
          </cpc:title-part>
          <cpc:explanation>
            <cpc:text scheme="ipc">
menstruation tables
<cpc:class-ref scheme="cpc">G06C3/00</cpc:class-ref>
            </cpc:text>
          </cpc:explanation>
          </cpc:title-part>
          <cpc:title-part>
            <cpc:text scheme="ipc">Throat striking implements</cpc:text>
          </cpc:title-part>
        </cpc:class-title>
      </ops:classification-statistics>
    </ops:search-result>
  </ops:classification-search>
</ops:world-patent-data>
```

```

    </cpc:title-part>
    </cpc:class-title>
    </ops:classification-statistics>
    <!-- 8 more classification-statistics elements removed for readability
of the example -->
    </ops:search-result>
  </ops:classification-search>
</ops:world-patent-data>

```

Range control

You can specify the number of search results you want to receive using the Range parameter. The default range of the CPC search response is 1-10. It can be changed by specifying an HTTP header parameter **X-OPS-Range=[begin_range-end_range]**.

For the same purposes, it is also possible to use the range query parameter by adding **&Range=[begin_range-end_range]** to the end of your URI (e.g. <http://ops.epo.org/rest-services/classification/cpc/search/?q=chemistry&Range=1-5>).

Example Request

```

GET http://ops.epo.org/rest-services/classification/cpc/search/?q=chemistry

Accept: application/ops+xml
X-OPS-Range: 1-5

```

Response

```

<ops:classification-search total-result-count="5" scheme-type="cpc">
  <ops:query syntax="CQL">titleandabstract = chemistry</ops:query>
  <!-- other elements removed for readability -->

```

Note the maximum range is 100.

3.6.4. Classification Mapping service

This service converts ECLA and CPC symbols to the corresponding symbols of the different classifications. Following conversion directions are supported:

Input format	Output format
ECLA	CPC
	IPC
CPC	ECLA
	IPC

Request structure

```
GET http://ops.epo.org/rest-services/classification/map/[input-format]/[classification-symbol]/[output-format]
```

```
Accept: application/ops+xml
```

Note: The service accepts only 'full' classification symbols, starting from Main group, like a01b1/00.

Response structure

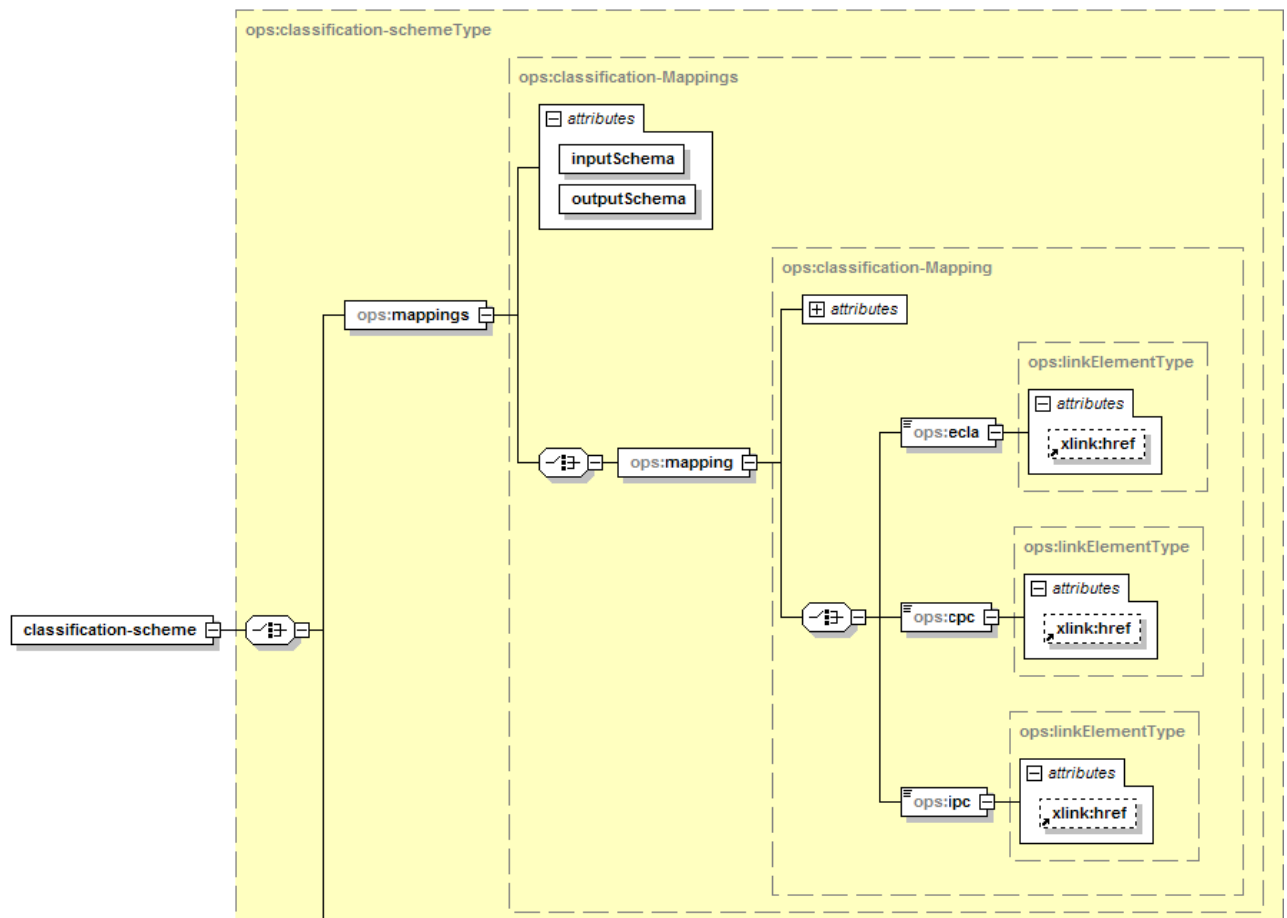


Figure 36

The structure shown in Figure 36 is part of the `ops` namespace and defined in the [ops.xsd](#) schema file.

Example Request

```
GET http://ops.epo.org/rest-services/classification/map/ecla/A61K9/00/cpc
```

```
Accept: application/ops+xml
```

Response

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:mappings inputSchema="ecla" outputSchema="cpc">
      <ops:mapping additional-only="false">
        <ops:ecla>A61K9/00</ops:ecla>
        <ops:cpc xlink:href="classification/cpc/A61K9/00">A61K9/00</ops:cpc>
      </ops:mapping>
    </ops:mappings>
  </ops:classification-scheme>
</ops:world-patent-data>
```

Note: The CPC to ECLA transformation can be ambiguous and it's not possible to determine automatically for some cases what kind of output is required (INVENTION or ADDITIONAL). To resolve that kind of ambiguity there is an optional request parameter *additional* in order to specify that additional classification symbol transformation is requested. The default request (without *additional* parameter specified) is invention type request.

Example Request

```
GET http://ops.epo.org/rest-
services/classification/map/cpc/A01D2085/008/ecla?additional

Accept: application/ops+xml
```

Response

```
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:cpc="http://www.epo.org/cpcexport">
  <ops:classification-scheme>
    <ops:mappings inputSchema="cpc" outputSchema="ecla">
      <ops:mapping additional-only="true">
        <ops:ecla>K01D85/00F2</ops:ecla>
        <ops:cpc
xlink:href="classification/cpc/A01D2085/008">A01D2085/008</ops:cpc>
      </ops:mapping>
    </ops:mappings>
  </ops:classification-scheme>
</ops:world-patent-data>
```

3.7. CPC-International (CPC-I)

Prior to the introduction of CPC International (August 2019) the format of marshalling of B28B1/29 <http://ops.epo.org/3.2/rest-services/published-data/publication/docdb/EP.1000000.A1/biblio> was

```
<patent-classifications>
  <patent-classification sequence="1">
```

```

<classification-scheme office="" scheme="CPC"/>
<section>B</section>
<class>28</class>
<subclass>B</subclass>
<main-group>1</main-group>
<subgroup>29</subgroup>
<classification-value>I</classification-value>
</patent-classification>
</patent-classifications>

```

CPC-I has introduced a new way of expressing international classification symbols B28B1/29 (CN, EP, KR, US) followed by A01B1/00 (EP)

```

<patent-classifications>
  <patent-classification sequence="1">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>CN</generating-office>
  </patent-classification>
  <patent-classification sequence="2">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>EP</generating-office>
  </patent-classification>
  <patent-classification sequence="3">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>KR</generating-office>
  </patent-classification>
  <patent-classification sequence="4">

```

```

<classification-scheme office="EP" scheme="CPCI"/>
<section>B</section>
<class>28</class>
<subclass>B</subclass>
<main-group>1</main-group>
<subgroup>29</subgroup>
<classification-value>I</classification-value>
<generating-office>US</generating-office>
</patent-classification>
<patent-classification sequence="5">
  <classification-scheme office="EP" scheme="CPCI"/>
  <section>A</section>
  <class>01</class>
  <subclass>B</subclass>
  <main-group>1</main-group>
  <subgroup>00</subgroup>
  <classification-value>I</classification-value>
  <generating-office>US</generating-office>
</patent-classification>
</patent-classifications>

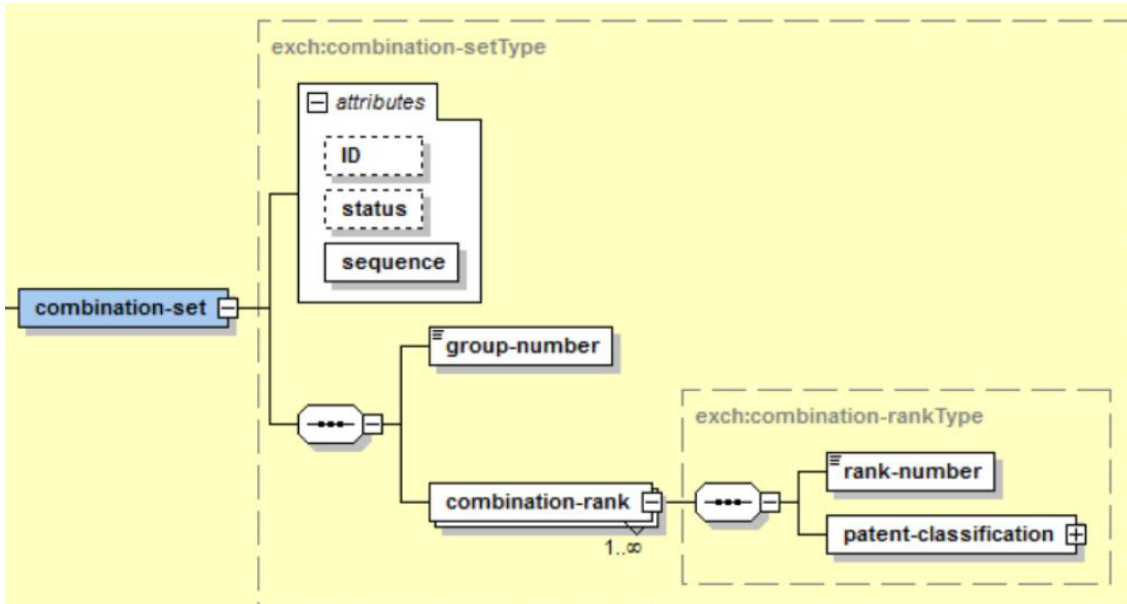
```

The following aspects of marshalling of CPC classification has changed:

- Attributes of CPCI `classification-scheme` are always `office="EP" scheme="CPCI"`
- `generating-office` element contains a country code of CPCI
- A compact line B28B1/29 (EP, CN, KR, US) format is expanded to 4 XML elements
- in CPC-I symbols country codes are sorted alphabetically with EP always coming first.
- Sequence number in scope of one CPC international class increments. See example above.
- OPS service does not do any de-duplication, client applications can implement de-duplication if necessary.

3.7.1. Combination- sets

Structure of `combination-set` ("combi-set") element:



Prior to CPC International combi-set marshalling, examples:

<https://ops.epo.org/3.2/rest-services/published-data/publication/docdb/DK.2197935.T3/biblio>

<https://ops.epo.org/3.2/rest-services/published-data/publication/docdb/DE.202015009630.U1/biblio>

<https://ops.epo.org/3.2/rest-services/published-data/publication/docdb/PL.2756098.T3/biblio>

```

<combination-set sequence="1">
  <group-number>1</group-number>
  <combination-rank>
    <rank-number>1</rank-number>
    <patent-classification>
      <classification-scheme office="" scheme="CPC"/>
      <classification-symbol>C08F220/06</classification-symbol>
      <classification-value>A</classification-value>
    </patent-classification>
  </combination-rank>
  <combination-rank>
    <rank-number>2</rank-number>
    <patent-classification>
      <classification-scheme office="" scheme="CPC"/>
      <classification-symbol>C08F2222/1013</classification-symbol>
      <classification-value>A</classification-value>
    </patent-classification>
  </combination-rank>
  <combination-rank>
    <rank-number>3</rank-number>
    <patent-classification>
      <classification-scheme office="" scheme="CPC"/>

```

```

        <classification-symbol>C08F2222/108</classification-symbol>
        <classification-value>A</classification-value>
    </patent-classification>
</combination-rank>
<combination-rank>
    <rank-number>4</rank-number>
    <patent-classification>
        <classification-scheme office="" scheme="CPC"/>
        <classification-symbol>C08F2222/1026</classification-symbol>
        <classification-value>A</classification-value>
    </patent-classification>
</combination-rank>
</combination-set>

```

After introduction of CPC-I, international combi-set CPC (CL) are expressed as follows:

- C12Q1/6818, C12Q2525/117, C12Q2535/131, C12Q2537/101, C12Q2563/107, C12Q2565/1015, C12Q2565/107, INV (EP, US)
- C12Q1/6827, C12Q2525/117, C12Q2563/107, C12Q2565/1015, C12Q2565/107, INV (EP, US)

```

<combination-set sequence="1">
    <group-number>1</group-number>
    <combination-rank>
        <rank-number>1</rank-number>
        <patent-classification>
            <classification-scheme office="EP" scheme="CPCI"/>
            <classification-symbol>C12Q1/6818</classification-symbol>
            <classification-value>I</classification-value>
            <generating-office>EP</generating-office>
        </patent-classification>
    </combination-rank>
    <combination-rank>
        <rank-number>2</rank-number>
        <patent-classification>
            <classification-scheme office="EP" scheme="CPCI"/>
            <classification-symbol>C12Q2525/117</classification-symbol>
            <classification-value>I</classification-value>
            <generating-office>EP</generating-office>
        </patent-classification>
    </combination-rank>
    <combination-rank>
        <rank-number>3</rank-number>

```

```

    <patent-classification>
      <classification-scheme office="EP" scheme="CPCI"/>
      <classification-symbol>C12Q2535/131</classification-symbol>
      <classification-value>I</classification-value>
      <generating-office>EP</generating-office>
    </patent-classification>
  </combination-rank>
  ...
</combination-set>

```

Following aspects of marshalling of CPC combi-set symbols has changed:

- Attributes of CPCI `classification-scheme` are always `office="EP" scheme="CPCI"`
- `generating-office` element contains a country code of CPCI
- A compact line (e.g. C08F2222/1013 INV (EP,CN)) format is expanded to 2 `combination-rank` XML elements
- `rank-number` reflects the number in a sequential number of a classification combi-set therefore for multi-country classes it increments, see example above.

3.7.2. CPC-I optional condensed format

Additionally to the default verbose format shown above, OPS gives the possibility to retrieve the response with a condensed CPC-I presentation.

- Condensed CPC-I format is requested by specifying the value `cpci=condensed` within the optional `cpci` query parameter:

```

GET /rest-services/published-
data/publication/docdb/EP.1000000.A1/biblio?cpci=condensed
Accept: application/json

```

- Condensed mode is supported by all "biblio" services
- Absence or different value of `cpci` query parameter will lead to verbose response format.

Verbose format (default)

```

<patent-classifications>
  <patent-classification sequence="1">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>EP</generating-office>
  </patent-classification>
  <patent-classification sequence="2">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>

```

```

    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>CN</generating-office>
  </patent-classification>
  <patent-classification sequence="3">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>KR</generating-office>
  </patent-classification>
  ...
</patent-classifications>

```

Condensed format:

```

<patent-classifications>
  <patent-classification sequence="1">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>B</section>
    <class>28</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>29</subgroup>
    <classification-value>I</classification-value>
    <generating-office>EP,CN,KR,US</generating-office>
  </patent-classification>
  <patent-classification sequence="2">
    <classification-scheme office="EP" scheme="CPCI"/>
    <section>A</section>
    <class>01</class>
    <subclass>B</subclass>
    <main-group>1</main-group>
    <subgroup>00</subgroup>
    <classification-value>I</classification-value>
    <generating-office>US</generating-office>
  </patent-classification>
</patent-classifications>

```

4. APPENDIX

- 3.1. Published-data services
- 3.2. Family service
- 3.3. Number-service
- 3.4. Register service
- 3.5. Legal service
- 3.6. Classification services

[top of document](#)

4.1. Relevant acronyms and abbreviations

Abbreviation or acronym	Description
CQL	Contextual or Common Query Language
DTD	Document Type Definition
ECLA	European Classification system
HTTP	Hypertext Transfer Protocol
IPC ^{digit}	International Patent Classification, <i>digit</i> -th release (if skipped, 8th is assumed)
MIME	Multipurpose Internet Mail Extensions
PNG	Portable Network Graphics
REST	Representational State Transfer
ST.36	WIPO Standard 36: Processing of patent information using XML
TIFF	Tagged Image File Format
W3C	World Wide Web Consortium
WADL	Web Application Description Language
WS- <i>term</i>	A <i>term</i> in context of the Web Services (Architecture)
XML	Extensible Markup Language
XSD	XML Schema Definition

Table 27

4.2. CQL index catalogue

The following is a guide for using CQL in published-data service bibliographic search. Note, CQL details in this section **ARE NOT VALID** for Register service (see the CQL details for [Register search in section 3.4.2](#))

CQL usage in OPS can be summarised in the following statements:

1. A CQL query is either a search clause or multiple search clauses connected by a Boolean operator.
2. A search clause consists of an index, relation and search term or a search term alone, where both might be enclosed within parentheses.
3. If the index is missing then the search clause must not have the relation, it is either both of them or none of them.
4. If the index and a relation is missing then an equality relation is assumed and the index is determined based on the following rules:
 - If a search term is a 2 letter ISO country code, the `num` index is assumed.
 - If a search term matches one of the date formats mentioned below for `pd` index then the `pd` index is assumed. The 4-digits year (`YYYY`) should be within the range 1800-2999, both a month number (`MM`) and a day number (`dd`) are having leading zero if necessary.
 - If a search term matches one of the following patterns: `x`, `xdd`, `xddw`, `xddwd`, `xddwdd`, `xddwdd/h`, the `cl` index is assumed: `x` refers to one letter (either upper or lower case) of a classification group within the range `a-h` or `y`, `d` is a digit, `w` refers to any alphanumeric character and `h` is a hexadecimal number up to 6 digit long.
 - If a search term matches `\\w{2,4}\\d{1,}[a-zA-Z]?\\d?` or is composed of digits only, then the `num` index is assumed.
 - If a search term is composed of letters only and starts with capital letter then the `ia` index is assumed.
 - Otherwise `txt` index is assumed.
5. A relation means either:
 - an equality relation which exact definition varies depend on a kind of a search term and index:
 - when index values fall within the range (like publication date) and search term is a list of exactly 2 words it is assumed that equality relation is within relation,
 - otherwise, equality means that index value and search term are identical,
 - an order relation for comparable terms: less (`<`), greater (`>`), equal (`=` or `==`), less or equal (`<=`), greater or equal (`>=`),

- an relational qualifier (`=/low`, `=/high`, `=/same`) can be used with CPC classification indices only,
- an occurrence relation on search term which is a list of words:
 - for a terms which are list of exactly 2 words with values that fall within the range, like dates or numerical values, where words are respectfully the beginning and the end of a range,
 - `any` or `all` relation for a terms which are list of words, where `any` means any of these words and `all` means all of these words,

6. A search term is either a single word or a list of words enclosed within double quotes assumed to be separated by a comma or whitespace if the `within` operator is used.

7. A word might contain truncation characters:

- unlimited truncation (`*`) which represents a string of any length including any character,
- limited truncation (`?`) which represents any character or no character,
- masking truncation (`#`) which represents any character which is mandatory present,
- it is possible to use truncation at the beginning of a word only in title and abstract indices.

8. A Boolean operator is one of logical operators: `and`, `or`, `not` or a proximity operator. Mathematically, operator `not` is not an unary operator and should be understood like "and not" in a common sense, and this operator may not begin the expression (`not pd=2010` is illegal).

9. A proximity operator `prox` is either checking a co-occurrence of two search terms in scope of a defined unit of text (`word`, `sentence` or `paragraph`) or checking the distance between those two search terms with or without the respect of its order. Both terms must be in scope of the same index.

10. Relation qualifiers (`=/low`, `=/high`, `=/same`) can only be used with single CQL CPC classification index and can be combined with the other CQL operators and qualifiers within the same query.

Table 28

Catalogue of supported CQL queries for OPS Published-data service

Since the information about publications is structured (see [exchange document structure in the section 2.2.2 Common response structures](#)) the following catalogue of CQL queries has been designed. The column titled 'Single correspondence', means "corresponds to a single piece of information" or exactly one element from the exchange document structure.

Index name	Single correspondence?	Description
title, ti	yes	the publication title in English
abstract, ab	yes	the publication abstract in English
titleandabstract, ta	no	the publication title or abstract in English
inventor, in	yes	an inventor name
applicant, pa	yes	an applicant name
inventorandapplicant, ia	no	a name of an inventor or an applicant
publicationnumber, pn	yes	the publication number in any format
spn	yes	the publication number in epodoc format
applicantnumber, ap	yes	the application number in any format
sap	yes	the application number in epodoc format
prioritynumber, pr	yes	the priority number
spr	yes	the priority number in epodoc format
num	no	the publication, application or a priority number in any format
publicationdate, pd	yes	the publication date in one of the following formats: <ul style="list-style-type: none"> • YYYY • YYYYMM • YYYYMMdd • YYYY-MM • YYYY-MM-dd • MM/YYYY • dd/MM/YYYY • MM.YYYY • dd.MM.YYYY
citation, ct	yes	a cited document number
ex	yes	a cited document number during the examination
op	yes	a cited document number during the opposition

Index name	Single correspondence?	Description
rf	yes	a cited document number provided by the applicant
oc	yes	another cited document number
famn	yes	a simple family identifier
cpc cpci cpca cpcc	yes	an CPC classification CPC invention classifications CPC additional classifications CPC linked symbols confirmed + National Office
ipc, ic	no	any IPC1-8 class
ci	yes	IPC8 core invention class
cn	yes	IPC8 core additional class (non-invention)
ai	yes	IPC8 advanced invention class
an	yes	IPC8 advanced additional class (non-invention)
a	no	any IPC8 advanced class
c	no	any IPC8 core class
cl	no	an CPC or IPC1-8 class
txt	no	publication title or abstract, or inventor/applicant name

Table 29

CQL examples

The following examples illustrate the CQL usage in OPS. Please note, that some special characters must be replaced when placed inside XML document, particularly the 'less than' symbol <, must be replaced with < (including the semicolon) and the 'greater than' symbol >, must become > (including the semicolon).

Query	Description of result
ti all "green energy"	All publications where "green" and "energy" are found among all titles. <i>Since the ti index checks title in all languages, it is possible, but unlikely, that one word will be found in English title and another in the German (etc.) title.</i>
ti="green energy technology"	All publications where a title contains exact words/phrase "green energy technology"
ti=green prox/unit= paragraph ti=energy	All publications where a title contains "green" and "energy" together in a same paragraph.
pd within "20051212 20051214"	All publications published 12, 13 or 14 December 2005
pd="20051212 20051214"	Same as above
ia any "John Smith"	All publications with "John" and/or "Smith" in the applicant/inventor name
pn=EP and pr=GB	All publications published by EPO (EP country code in the publication number) having a priority document from United Kingdom (GB country code in the priority number)
ta=green prox/distance<=3 ta=energy	All publications having words "green" and "energy" in either title or abstract separated by at most 3 words from each other,
ta=green prox/distance<=2/ordered= true ta=energy	All publications having a word "green" followed by a word "energy" in either title or abstract separated by at most 2 words,
(ta=green prox/distance<=3 ta=energy) or (ta=renewable prox/distance<=3 ta= energy)	All publications having either "green" or "renewable" separated by at most 3 words from a word "energy" in either title or abstract, in any order.
pa all "intelligence agency atomic" and JP	All publications with a publication/application/priority document issued by Japan (having a JP country code in a document number) having "intelligence", "agency", "atomic" words in an applicant name

Query	Description of result
pa all "intelligence agency atomic" and JP and pd>2000	See above, but published after 2000.
pd < 18000101	All publications published before 18th century.
ta=synchroni#ed	Allows to look for "synchronized" or "synchronised" word in abstract/title
EP and 2009 and Smith	All publications with application/publication/priority document issued by EPO (having EP country code), published in 2009, having Smith as an applicant or inventor
cpc=/low A01B ct=EP1027777 txt all "vision technologies" cpc=(C08F prox/unit=sentence (US, EP)) cpc=(A01B1/00/low prox/unit=sentence (EP, US))	All publications classified with A01B and all subclasses All publications citing EP1027777 All publications where words "vision" and "technology" can be found either in title, abstract, inventor or applicant name All publications where the CPCI contains symbol C08F and has been classified both by US and EP Offices. All publications where the CPCI contains related class codes having lower classification hierarchy level regarding A01B1/00 and has been classified both by US and EP Offices.

Table 30

Extended CQL examples

OPS also supports so-called extended CQL language (see "[Smart Search](#)").

Input Query	Standard CQL equivalent
G08B25 H04L63	cl=G08B25 and cl=H04L63
G08B25 H04L63 title=grid	cl=G08B25 and cl=H04L63 and title=grid
Siemens EP 200701	inventorandapplicant=Siemens and num=EP and publicationdate=200701
20070101:20070115	publicationdate within "20070101 20070115"

4.3. The epodoc publication format

The number is in this format: CCNNNNNNNNNNNN (K)

Key:

CC: an ISO two-letter country code

N(N...): can be up to 12 digits (EP is 7), **NO spaces**

(K): **optional** attached kind code one-letter

Example:

```
<document-id document-id-type="epodoc">
  <doc-number>EP1915004</doc-number>
  <date>20080423</date>
</document-id>
```

For kind code (K) is:

- if it's A(followed by the number), letter A is never attached
- if it's D(followed by the number) to Z(followed by the number), the first letter is always attached:
- if it's B(followed by the number) or C(followed by the number), the first letter may be attached, *IF* it is necessary to differentiate documents belonging to overlapping series of numbers.

Examples:

JP2000177507 (**docdb** kind code A, is not "visible" in **epodoc** format)

JP3000014B (**docdb** kind code B1, only B is attached in **epodoc** format)

JP3000014U (different invention than the previous, **docdb** kind code U is attached in **epodoc** format)

CN100520025C (**docdb** kind code C is attached in **epodoc** format)

DE6610524U (**docdb** kind code U is attached in **epodoc** format)

KR200142084Y (**docdb** kind code Y1, only Y is attached in **epodoc** format)

4.4. Expressing PCT application numbers in docdb format

The format has changed through the years:

- Until 1.1.2004: CCyynnnnnW
- After 1.1.2004: CCccyynnnnnW

Key:

- CC=country code, which references where the filing took place;
(**Note**, IB=International Buro.)
- cc=century (20th) - this part was added from 1.1.2004 and onwards
- yy=year
- nnnnnn=sequential number (6 digits; 5 digits before 2004)
- W= **mandatory** application kind code

Example:

PCT/GB02/04635 translates in docdb to GB 0204635 W

Please refer to the number-service section for an [example on how to convert PCT application numbers to docdb format](#).

4.5. Number-service Use Cases

As described in chapter 3.3, the number-service provides the functionality to transform application, publication and priority numbers from one [input format](#) to another format. The purpose of this chapter is to give you a "real life" example of when and how to use this service.

As OPS provides access to data that is held inside the EPO, the OPS services (except the number-service) require the `input` to be in one of the two EPO formats: `epodoc` or `docdb`. The most common use case of the number-service is thus to transform numbers that are given in domestic formats of different countries to the EPO formats that can then be used for receiving published, family, register, legal or classification data.

Assuming the following scenario: starting with an original document for a specific patent application, you want to receive data from OPS related to this document:



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) DE 10 2006 032 425 A1 2007.01.18

(12)

Offenlegungsschrift

(21) Aktenzeichen: 10 2006 032 425.0

(22) Anmeldetag: 13.07.2006

(43) Offenlegungstag: 18.01.2007

(51) Int. Cl.?: **G11B 15/60** (2006.01)

G11B 23/12 (2006.01)

G11B 5/02 (2006.01)

(30) Unionspriorität:
11/183,143 14.07.2005 US

(71) Anmelder:
Quantum Corp., San Jose, Calif., US

(74) Vertreter:
WUESTHOFF & WUESTHOFF Patent- und
Rechtsanwälte, 81541 München

(72) Erfinder:
McCormack, Patrick, Costa Mesa, Calif., US

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: **Externes Tischdock für eine Datenspeichereinheit auf Kassettenbasis**

(57) Zusammenfassung: Eine externe Andockstation (100) für eine entfernbare Datenspeichereinheit (90) gestattet dieser die Kommunikation mit einem Computersystem über eine standardmäßige Schnittstellenverbindung wie etwa einen USB-Anschluss (108). Bei einer Ausgestaltung verriegelt die Andockstation die Datenspeichereinheit mittels eines Verriegelungselements in einem Verriegelungs- oder Schließzustand. Die Andockstation (100) umfasst eine motorische, über eine Zahnradanordnung angetriebene Hebelanordnung, welche die Datenspeichereinheit entriegelt und gleichzeitig aus der Andockstation auswirft. Eine Feder spannt das Verriegelungselement vor, um die Andockstation für die Aufnahme der Datenspeichereinheit in einem Offenzustand zu halten. Nach Drücken einer Auswurf Taste (103) durch den Benutzer erfolgt eine Verzögerung des Auswerfens der Datenspeichereinheit solange, bis ein Festplattenlaufwerk nicht mehr in Betrieb ist. Auf diese Weise wird eine mögliche Beschädigung des Laufwerksmediums und/oder von Lese-/Schreibköpfen vermieden.

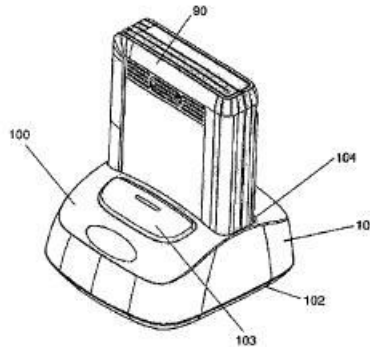


Figure 37

As you can see there are three different numbers on the first page of this original document:

1. Publication number (10): DE 10 2006 032 425 A1
2. Application number (21): 10 2006 032 425.0
3. Priority number (30): 11/183,143 US

Note, for a full list of the INID codes inside of patent documents, see the [documentation of the WIPO ST9](#).

The priority – what other documents are related to it?

In our use case we want to receive family information for the `priority` of this document.

To achieve this we have to first convert the given number in `original` format to the `docdb` format:

```
GET http://ops.epo.org/rest-services/number-service/priority/original/US.\(11/183,143\).20050714/docdb
```

```
Accept: application/ops+xml
```

Response:

Number Service (>)

inputFormat: original - outputFormat: docdb

Meta:	Status:	BRW003 BRW008
	Version:	10.09.93
Input:	US 11/183,143 20050714	
Output:	US 18314305 A 20050714	

Figure 38

The number service response includes the priority number in `docdb` format which can then be used as `input` for the family service:

```
GET http://ops.epo.org/rest-services/family/priority/docdb/US.18314305.A
```

```
Accept: application/ops+xml
```

Response:

Patent Family

legal: false

Meta:	Elapsed Time:	126
DOCDB:	Country:	US
	Doc-number:	18314305
	Kind:	A

Family Member (1):	Publication Reference:	DOCDB:	Country:	DE
			Doc-number:	102006032425
			Kind:	A1
			Date:	20070118
	EPODOC:		Doc-number:	DE102006032425
			Date:	20070118
	Application Reference:	DOCDB:	Country:	DE
			Doc-number:	102006032425
			Kind:	A
			Date:	20060713
	Priority Claim:	DOCDB:	Country:	US
			Doc-number:	18314305
			Kind:	A
			Date:	20050714

Figure 39

The family contains 6 members that claim the US priority.

Retrieving the bibliographic data for the German publication

A second use case is to retrieve the bibliographic data for the German publication number (DE 10 2006 032 425 A1) . To achieve this we again have to transform the original number using the number-service. This time we request `epodoc` as the output format:

```
GET http://ops.epo.org/rest-services/number-service/publication/original/DE.\(10\_2006\_032\_425\).A1.20070208/epodoc
```

```
Accept: application/ops+xml
```


Response:**Number Service (>)**

inputFormat: original - outputFormat: epodoc

Meta:	Version:docdb/epodoc: 2010.03.19
	Version:original/docdb: 10.09.93
Input:	DE 10 2006 032 425 A1 20070208
Output:	DE 102006032425 A1 20070208

Figure 40

The publication number in `epodoc` format can then be used as input for the biblio retrieval:

```
GET http://ops.epo.org/rest-services/published-data/publication/epodoc/DE102006032425/biblio
```

```
Accept: application/ops+xml
```

Response:**DE 102006032425 (A1) –****Bibliographic Data**

system: ops.epo.org - family-id: 36888262 - country: DE - doc-number: 102006032425 - kind: A1

Publication Reference: DOCDB:	Country: DE
	Doc-number: 102006032425
	Kind: A1
	Date: 20070118
EPODOC:	Doc-number: DE102006032425
	Date: 20070118
Classifications:	IPC: G11B 15/60 (G11B 15/60 A1); G11B 5/02 (G11B 5/02 A1); G11B 23/12 (G11B 23/12 A1)
	EC: G06F 1/16P6; G11B25/04R; G11B33/02A1; G11B33/12B1
Application Reference: EPODOC:	Doc-number: DE200610032425
	Date: 20060713
	ORIGINAL: Doc-number: 102006032425
Priority Claims:	EPODOC: 1) Doc-number: US20050183143
	Date: 20050714
	1) Doc-number: 11/183,143
	ORIGINAL: 1) Doc-number: US20050183143
	Date: 20050714
	1) Doc-number: 11/183,143

Figure 41

Note this screenshot only shows the first part of the output.

4.6. Further reading

It is strongly recommended to review these resources to fully understand the concepts and environment of OPS.

Web Application Description Language

<http://www.w3.org/Submission/wadl/>

HTTP protocol

<http://www.w3.org/Protocols/rfc2616/rfc2616.html>

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

<http://www.w3.org/TR/xmlschema11-1/>

XML Schema Part 2: Datatypes Second Edition

<http://www.w3.org/TR/xmlschema-2/>

CQL: Common Query Language, Version 1.1

<http://www.loc.gov/standards/sru/cql/index.html>

RFC 6749 – The OAuth 2.0 Authorization Framework

<http://tools.ietf.org/html/rfc6749>