Patent Information

Open Patent Services RESTful Web Services
Reference Guide

Version 1.3.8
## Revision history

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Description</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/03/2011</td>
<td>1.0.0</td>
<td>Content development; master draft consultation version</td>
<td>Alexander Khval, Patricia Passarelli, Silke Szymura</td>
</tr>
<tr>
<td>16/05/2011</td>
<td>1.0.1</td>
<td>Update according to changes in the new OPS release</td>
<td>Silke Szymura</td>
</tr>
<tr>
<td>20/06/2011</td>
<td>1.0.2</td>
<td>Update according to changes in the next OPS release</td>
<td>Silke Szymura</td>
</tr>
<tr>
<td>12/12/2011</td>
<td>1.0.3</td>
<td>Update to Family priority-claim response</td>
<td>Dominic Afriat</td>
</tr>
<tr>
<td>01/03/2012</td>
<td>1.0.4</td>
<td>Update according to 3.0 OPS release</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>19/11/2012</td>
<td>1.1</td>
<td>Updated to reflect CPC changes</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>30/05/2013</td>
<td>1.2</td>
<td>Updated for OPS 3.1 and user Registration</td>
<td>Dominic Afriat</td>
</tr>
<tr>
<td>10/10/2013</td>
<td>1.2.8</td>
<td>Corrections to some Registered user response behaviour</td>
<td>Dominic Afriat</td>
</tr>
<tr>
<td>08/11/2013</td>
<td>1.2.9</td>
<td>Number Service status codes update</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>18/11/2013</td>
<td>1.2.10</td>
<td>Timezone specified</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>16/04/2014</td>
<td>1.2.11</td>
<td>Quota error messages updated</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>05/09/2014</td>
<td>1.2.12</td>
<td>Fixed links. Improved information related to image services.</td>
<td>Dmitry Katsubo</td>
</tr>
<tr>
<td>15/09/2014</td>
<td>1.2.13</td>
<td>CQL queries are updated</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>28/01/2015</td>
<td>1.2.14</td>
<td>Corrected the table with valid combinations of endpoints. CA authority supports fulltext.</td>
<td>Alexander Khval, Dmitry Katsubo</td>
</tr>
<tr>
<td>24/04/2015</td>
<td>1.2.15</td>
<td>Register CQL specification link is corrected</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>15/01/2016</td>
<td>1.2.16</td>
<td>Added date formats available in CQL. Added extended CQL section. Minor CQL fixes in relation to restrictions and syntax. Added “cpcc” index.</td>
<td>Dmitry Katsubo</td>
</tr>
<tr>
<td>25/01/2016</td>
<td>1.2.17</td>
<td>Removed <code>&lt;ops:meta elapsed-time=&quot;...&quot;</code>&gt;. Added INPADOC family total-result-count.</td>
<td>Dmitry Katsubo</td>
</tr>
<tr>
<td>16/02/2016</td>
<td>1.2.18</td>
<td>Added new error message RequestTimeout.</td>
<td>Aliaksei Lizunou</td>
</tr>
<tr>
<td>15/04/2016</td>
<td>1.3.0</td>
<td>Register UPP information</td>
<td>Alexander Khval</td>
</tr>
<tr>
<td>10/06/2016</td>
<td>1.3.1</td>
<td>Reworked constituents table. Updated CQL examples according to user's request.</td>
<td>Dmitry Katsubo</td>
</tr>
<tr>
<td>22/05/2017</td>
<td>1.3.2</td>
<td>Hyperlinks update and encryption protocols</td>
<td>Aliaksei Lizunou</td>
</tr>
<tr>
<td>07/08/2017</td>
<td>1.3.3</td>
<td>Update the epodoc publication format</td>
<td>Aliaksei Lizunou</td>
</tr>
<tr>
<td>26/09/2017</td>
<td>1.3.4</td>
<td>Removed all UPP and anonymous access references. Updated description of developer’s area.</td>
<td>Aliaksei Lizunou</td>
</tr>
<tr>
<td>04/12/2017</td>
<td>1.3.5</td>
<td>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.</td>
<td>Aliaksei Lizunou</td>
</tr>
<tr>
<td>15/01/2018</td>
<td>1.3.6</td>
<td>Fixed inaccuracy in a value of header Content-Type.</td>
<td>Aliaksei Lizunou</td>
</tr>
<tr>
<td>04/05/2018</td>
<td>1.3.7</td>
<td>Removed information about weekly free quota for registered users.</td>
<td>Aliaksei Lizunou</td>
</tr>
</tbody>
</table>
Updated authentication response view.

Aliaksei Lizunou
# 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 .........................................................29
   2.3. Registration & OPS Fair use policy .................................30
      2.3.1. User Registration for access .......................................31
      2.3.2. Authentication & Access Token handling ....................32
      2.3.3. Dynamic fair use monitoring .......................................39
      2.3.4. Data usage API ........................................................43
      2.3.5. Encryption protocols ................................................46

3. **OPS SERVICES** .................................................................47
   3.1. PUBLISHED-DATA SERVICES ......................................47
      3.1.1. Bibliographic data ....................................................50
      3.1.2. Fulltext inquiry and retrieval including description or claims ..60
      3.1.3. Images inquiry and retrieval .....................................66
      3.1.4. Equivalents only or with abstract, biblio, full-cycle, images 72
   3.2. Family service ..............................................................81
      3.2.1. Family data (without constituents) ............................84
      3.2.2. Family data with biblio ............................................85
      3.2.3. Family data with legal ..............................................86
      3.2.4. Family data with biblio and legal ..............................88
   3.3. Number-service ............................................................90
      3.3.1. From docdb to epodoc .............................................97
3.3.2. From original to docdb (JPO example) ....................................................... 97
3.3.3. From docdb to original (JPO example) .................................................... 98
3.3.4. From original to epodoc (USPTO example) ............................................ 99
3.3.5. From epodoc to original ....................................................................... 100
3.3.6. From original to docdb (PCT example) ................................................ 100
3.4. Register service ...................................................................................... 102
  3.4.1. Register retrieval ............................................................................... 102
  3.4.2. Register search ............................................................................... 111
3.5. Legal service ............................................................................................ 115
3.6. Classification services .......................................................................... 119
  3.6.1. CPC Retrieval ................................................................................... 120
  3.6.2. CPC Media retrieval ......................................................................... 126
  3.6.3. CPC Search ..................................................................................... 127
  3.6.4. Classification Mapping service ......................................................... 130
4. APPENDIX ................................................................................................. 133
  4.1. Relevant acronyms and abbreviations .................................................. 133
  4.2. CQL index catalogue ........................................................................... 134
  4.3. The epodoc publication format ............................................................ 140
  4.4. Expressing PCT application numbers in docdb format ................. 141
  4.5. Number-service Use Cases ................................................................. 141
  4.6. Further reading ................................................................................... 146
1. INTRODUCTION

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.

<table>
<thead>
<tr>
<th>Concept</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patent application</td>
<td>The formal &quot;paperwork&quot; 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.</td>
</tr>
<tr>
<td>Patent publication</td>
<td>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.</td>
</tr>
<tr>
<td>Patent priority</td>
<td>The Paris Convention for the Protection of Industrial Property</td>
</tr>
<tr>
<td>Concept</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Concept</td>
<td>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).</td>
</tr>
<tr>
<td>Patent publication kind code</td>
<td>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).</td>
</tr>
<tr>
<td>Patent publication date</td>
<td>The date when a described invention becomes publicly available. Further details publication date (on Espacenet.com).</td>
</tr>
<tr>
<td>Patent application claims</td>
<td>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).</td>
</tr>
<tr>
<td>Patent citation(s)</td>
<td>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.</td>
</tr>
<tr>
<td>Simple patent family</td>
<td>All documents sharing exactly the same set of priorities. For further details see patent families (on <a href="http://www.epo.org">www.epo.org</a>).</td>
</tr>
<tr>
<td>Patent family</td>
<td>All documents sharing directly or indirectly at least one priority.</td>
</tr>
</tbody>
</table>

Table 1
1.3. EP Patent lifecycle and reference types

![Diagram summary]

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

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
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

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Usually http</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authority</td>
<td>Usually it’s the host name. The OPS production authority is ops.epo.org</td>
</tr>
<tr>
<td>Version</td>
<td>Currently, 3.0</td>
</tr>
<tr>
<td>Prefix</td>
<td>Always rest-services to distinguish the use of the OPS RESTful services</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Service reference-type</th>
<th>Service name; see [OPS services (section 3)].</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input-format</td>
<td>Mandatory for requests; format-specific; see below Input format (section 2.1.2).</td>
</tr>
<tr>
<td>Input:</td>
<td>Consists of 4 input parts: CC, number, [KC] and [date]. Note, in Classification service (section 3.6) the input is structured differently.</td>
</tr>
</tbody>
</table>

| 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; optional part of the epodoc input. (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; optional part of the input. |

| [endpoint] | Optional URI part; Used by published-data services and provides various parts of published data. Mandatory, for Number-service (section 3.3). |
| [constituent(s)] | Optional part(s) of the URI. 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](https://www.wipo.int). **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.

<table>
<thead>
<tr>
<th>Reference type/format</th>
<th>Example that appears on the actual document</th>
</tr>
</thead>
<tbody>
<tr>
<td>application/original</td>
<td>MD a 2005 0130</td>
</tr>
<tr>
<td>application/docdb</td>
<td>MD.20050130.A</td>
</tr>
<tr>
<td>application/epodoc</td>
<td>MD20050000130</td>
</tr>
</tbody>
</table>

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. 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

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

Note, in OPS these characters SHOULD NEVER BE ENCODED:

<table>
<thead>
<tr>
<th>Format</th>
<th>Examples</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>original</td>
<td>DE 20 2007 016 308.8</td>
<td>The original application number, as found in the original patent number.</td>
</tr>
<tr>
<td></td>
<td>OPS compliant presentation is with brackets (see above): DE.(20 2007 016 308.8)</td>
<td></td>
</tr>
</tbody>
</table>

Table 4

Table 5
### Format

<table>
<thead>
<tr>
<th>Format</th>
<th>Examples</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>And with the space escaped this number becomes: DE.(20%202007%20016 308.8)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>docdb</td>
<td>DE.202007016308.U</td>
</tr>
<tr>
<td></td>
<td>epodoc</td>
<td>DE200720016308U</td>
</tr>
</tbody>
</table>

**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). \(\text{Note, these constituents are explained in their respective sections.}\)
- 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.
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**

<table>
<thead>
<tr>
<th>OPS service</th>
<th>Possible value for Accept header</th>
</tr>
</thead>
<tbody>
<tr>
<td>Published-data</td>
<td>application/fulltext+xml</td>
</tr>
<tr>
<td>Published-data</td>
<td>application/exchange+xml</td>
</tr>
<tr>
<td>Published-data/images</td>
<td>image/pdf</td>
</tr>
<tr>
<td>Published-data/images</td>
<td>application/tiff</td>
</tr>
<tr>
<td>Published-data/images</td>
<td>image/png</td>
</tr>
<tr>
<td>Number-service</td>
<td>application/ops+xml</td>
</tr>
<tr>
<td>Classification/cpc</td>
<td>application/cpc+xml</td>
</tr>
<tr>
<td>CPC/images</td>
<td>image/gif</td>
</tr>
<tr>
<td>Register</td>
<td>application/register+xml</td>
</tr>
<tr>
<td>ALL OPS SERVICES that support XML</td>
<td>application/json</td>
</tr>
</tbody>
</table>
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/](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:**

```
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"
  }
}
```
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.:


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.

<table>
<thead>
<tr>
<th>Data Structure document</th>
<th>Namespace</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open Patent Services input and output schema</td>
<td>ops</td>
<td>This is the main namespace that references all of the others. Defines the structure of the XML output documents in OPS v.3.0. Additionally, it defines a structure used for exchanging legal information about the patent publication.</td>
</tr>
<tr>
<td>EPO Exchange Document</td>
<td>exch</td>
<td>Defines structures for exchanging patent documents, including abstracts, in line with WIPO standard ST.36. For more details see</td>
</tr>
</tbody>
</table>
## 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:**

![Application-reference response](image)

**Application reference attributes (used in OPS responses)**

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>doc-id</td>
<td>Unique patent identifier. Please refer to the <a href="https://www.epo.org/docdb/docs.asp">docdb user documentation</a> for more information.</td>
</tr>
<tr>
<td>is-representative</td>
<td>Identifies whether this application is the family representative in the EPO simple patent family, value = YES/NO</td>
</tr>
</tbody>
</table>

*Table 9*
Publication-reference response:

![Diagram of publication-reference response](image1)

Figure 4

Priority-claim response:

![Diagram of priority-claim response](image2)

Figure 5
Priority Claims elements (optional but used in OPS family responses)

<table>
<thead>
<tr>
<th>Attribute name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority-linkage-type</td>
<td>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.</td>
</tr>
<tr>
<td>priority-active-indicator</td>
<td>The DOCDB Active indicator, used to define Simple patent families. Please refer to the docdb user documentation for more information.</td>
</tr>
</tbody>
</table>

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):

![Diagram of document-id]

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:

```xml
<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">
```


Exchange Document

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

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):

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

Table 11
Figure 8
Some elements contained in the bibliographic-data element:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>publication-reference</td>
<td>Refers to the publication in every available format. To express different formats, document-id 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 yyyyMMdd, where yyyy refers to the publication year, MM to the publication month number (with the leading zero) and dd to the publication day number (with the leading zero).</td>
</tr>
<tr>
<td>classification-ipc</td>
<td>Provides the list of IPC1-7 classes the underlying publication has been classified into. 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.</td>
</tr>
<tr>
<td>classification-ipcr</td>
<td>Provides the list of IPC8 classes the underlying publication has been classified into, according to WIPO ST8.</td>
</tr>
<tr>
<td>patent-classification</td>
<td>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.</td>
</tr>
<tr>
<td>application-reference</td>
<td>Refers to the application document in every available format with the application date. In most cases, docdb and epodoc format are provided, original if available.</td>
</tr>
<tr>
<td>priority-claim</td>
<td>Contains a priority claim document reference in the available format(s) with the priority date. The sequence attribute refers to the order in which priorities are printed on the published document.</td>
</tr>
<tr>
<td>applicant (epodoc format)</td>
<td>Contains the applicant name(s) written in epodoc format.</td>
</tr>
<tr>
<td>applicant (original format)</td>
<td>Contains the applicant name(s) as it is written on the original paper document, if available.</td>
</tr>
<tr>
<td>inventor (epodoc format)</td>
<td>Contains the inventor name(s) written in epodoc format.</td>
</tr>
<tr>
<td>Element</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>inventor (original format)</td>
<td>Contains the inventor name(s) as it is written on the original paper document, if available.</td>
</tr>
<tr>
<td>invention-title</td>
<td>Contains the invention title in the original language and French, German or English, if available. The language code is provided within the lang attribute. There might be many invention-title elements which reflect the title structure, as printed on the publication document.</td>
</tr>
<tr>
<td>citation</td>
<td>Provides the publication references (in epodoc format) to the documents considered to be relevant during a single phase of a search/grant procedure. The sequence attribute refers to the order in which documents are printed on the search report document. For more information on see citations table below.</td>
</tr>
<tr>
<td>abstract</td>
<td>Contains the abstract paragraph, if available. The language code is provided within the lang attribute of the abstract element. There might be many abstract elements that reflect the abstract structure, as printed on the publication document.</td>
</tr>
</tbody>
</table>

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:

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 (nplcit).

patcit:
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):

<table>
<thead>
<tr>
<th>Letter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>particularly relevant if taken alone (- prejudicing novelty, from April 2011)</td>
</tr>
<tr>
<td>I</td>
<td>particularly relevant if taken alone – prejudicing inventive step (from April 2011)</td>
</tr>
<tr>
<td>Y</td>
<td>particularly relevant if combined with another document of the same</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>Phase name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>Cited during the description (application) phase</td>
</tr>
<tr>
<td>search</td>
<td>General Search phase; used when no further information available. Otherwise, use one of the concrete phases below:</td>
</tr>
<tr>
<td>International-search-report</td>
<td>Originates from International Search Report</td>
</tr>
<tr>
<td>supplementary-international-search-report</td>
<td>Originates from the international Supplementary Search Report</td>
</tr>
<tr>
<td>national-search-report</td>
<td>Originates from the national Search Report</td>
</tr>
<tr>
<td>supplementary-national-search-report</td>
<td>Originates from the national Supplementary Search Report</td>
</tr>
<tr>
<td>international-type-search-report</td>
<td>Originates from the international Type Search Report</td>
</tr>
<tr>
<td>examination</td>
<td>General Examination phase; use EXA when there is no further information available or one of the below if there is:</td>
</tr>
<tr>
<td>international-examination</td>
<td>Cited during PCT chapter II (preliminary</td>
</tr>
</tbody>
</table>
### Phase name

<table>
<thead>
<tr>
<th>Phase name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examination</td>
<td>Revealed during the examination phase; international examination</td>
</tr>
<tr>
<td>National-examination</td>
<td>Revealed during the national examination phase</td>
</tr>
<tr>
<td>Opposition</td>
<td>Revealed during the Opposition phase</td>
</tr>
<tr>
<td>Limitation</td>
<td>Revealed during the Limitation phase</td>
</tr>
<tr>
<td>Other</td>
<td>Other documents of interest (cited in comptes rendu but not in search report)</td>
</tr>
<tr>
<td>Unknown</td>
<td>No information available</td>
</tr>
</tbody>
</table>

*Table 14*

### 2.2.3. Error messages

There are some situations when the OPS may respond with an error message.

<table>
<thead>
<tr>
<th>HTTP code</th>
<th>Error code</th>
<th>Error occurrence</th>
</tr>
</thead>
<tbody>
<tr>
<td>404</td>
<td>CLIENT.InvalidReference</td>
<td>At least one reference in the request is invalid: &lt;DETAILS&gt;.</td>
</tr>
<tr>
<td>404</td>
<td>CLIENT.WrongReferenceFormatting</td>
<td>At least one reference in the request has a wrong value for data-format</td>
</tr>
<tr>
<td>413</td>
<td>CLIENT.AmbiguousRequest</td>
<td>The requested input was ambiguous¹.</td>
</tr>
<tr>
<td>404</td>
<td>SERVER.EntityNotFound</td>
<td>The requested input was ambiguous¹.</td>
</tr>
<tr>
<td>404</td>
<td>CLIENT.NotFound</td>
<td>Resource not found</td>
</tr>
<tr>
<td>400</td>
<td>CLIENT.InvalidQuery</td>
<td>The request was invalid.</td>
</tr>
<tr>
<td>400</td>
<td>CLIENT.CQL</td>
<td>The query provided is invalid, &lt;DETAILS&gt;.</td>
</tr>
<tr>
<td>405</td>
<td>CLIENT.MethodNotAllowed</td>
<td>Can't serve media types listed in the Accept header, &lt;DETAILS&gt;.</td>
</tr>
<tr>
<td>403</td>
<td>CLIENT.RobotDetected</td>
<td>Recent behaviour implies you are a robot. The server is at the moment busy to serve robots. Please try again later.</td>
</tr>
<tr>
<td>503</td>
<td>SERVER.LimitedServerResources</td>
<td>Please request bibliographic data in smaller chunks².</td>
</tr>
<tr>
<td>500</td>
<td>SERVER.DomainAccess</td>
<td>The request could not be processed.</td>
</tr>
</tbody>
</table>

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/docdb/EP.0536425/fulltext](http://ops.epo.org/rest-services/published-data/publication/docdb/EP.0536425/fulltext) will return the claims/description availability for the given number while [http://ops.epo.org/rest-services/published-data/publication/epodoc/EP.0536425/fulltext](http://ops.epo.org/rest-services/published-data/publication/epodoc/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.

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.
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.
- 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: y3AOQ86qwegU0QU69YGTJ4JGxUN8EVG
Consumer Secret Key: mXdr5WA7x9udmP
```

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.
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.
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:

<table>
<thead>
<tr>
<th>HTTPS</th>
<th>POST</th>
<th><a href="https://ops.epo.org/3.2/auth/accesstoken">https://ops.epo.org/3.2/auth/accesstoken</a></th>
</tr>
</thead>
</table>

**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

```
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"
}
```

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

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP Headers</td>
<td></td>
</tr>
<tr>
<td>Authorization: Bearer 4AWoepfVNgf09DRmimEnGdXcgoFU</td>
<td></td>
</tr>
</tbody>
</table>

**Error conditions**

**Spike/Quota Errors**

When

Returned when spike or quota limit has been reached.

**XML Response**

403 – FORBIDDEN

```xml
<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

```xml
<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

```xml
<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.

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.
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.
• 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. A 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=JBAPP_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:

<table>
<thead>
<tr>
<th>Status</th>
<th>Retrieval</th>
<th>Search</th>
<th>Inpadoc</th>
<th>Images</th>
<th>Other</th>
<th>Time out</th>
</tr>
</thead>
<tbody>
<tr>
<td>idle</td>
<td>green:200</td>
<td>green:30</td>
<td>green:60</td>
<td>green:200</td>
<td>green:1000</td>
<td></td>
</tr>
<tr>
<td>busy</td>
<td>green:100</td>
<td>green:15</td>
<td>green:45</td>
<td>green:100</td>
<td>green:1000</td>
<td></td>
</tr>
<tr>
<td>overloaded</td>
<td>green:50</td>
<td>green:5</td>
<td>green:30</td>
<td>green:50</td>
<td>green:1000</td>
<td></td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Status</th>
<th>Retrieval</th>
<th>Search</th>
<th>Inpadoc</th>
<th>Images</th>
<th>Other</th>
<th>Time out</th>
</tr>
</thead>
<tbody>
<tr>
<td>overloaded</td>
<td>green:50</td>
<td>green:5</td>
<td>black:0</td>
<td>green:50</td>
<td>green:1000</td>
<td>600000</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>t=0</th>
<th>Status</th>
<th>Retrieval</th>
<th>Search</th>
<th>Inpadoc</th>
<th>Images</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>idle</td>
<td>green:200</td>
<td>green:30</td>
<td>green:60</td>
<td>green:200</td>
<td>green:1000</td>
</tr>
<tr>
<td>t=1</td>
<td>busy</td>
<td>green:100</td>
<td>green:15</td>
<td>green:45</td>
<td>green:100</td>
<td>green:1000</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>t=20</th>
<th>Status</th>
<th>Retrieval</th>
<th>Search</th>
<th>Inpadoc</th>
<th>Images</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>overloaded</td>
<td>green:50</td>
<td>green:5</td>
<td>red:30</td>
<td>green:50</td>
<td>green:1000</td>
</tr>
<tr>
<td>t=21</td>
<td>overloaded</td>
<td>green:50</td>
<td>green:5</td>
<td>green:30</td>
<td>green:50</td>
<td>green:1000</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>Throttle</th>
<th>Rest service URI</th>
<th>SOAP service</th>
</tr>
</thead>
<tbody>
<tr>
<td>search</td>
<td>/published-data/search/*</td>
<td>biblio-search</td>
</tr>
<tr>
<td>retrieval</td>
<td>/published-data/*/</td>
<td>biblio-retrieval</td>
</tr>
<tr>
<td>inpadoc</td>
<td>/family/*</td>
<td>family-*-retrieval</td>
</tr>
<tr>
<td>inpadoc</td>
<td>/legal/*</td>
<td>legal-retrieval</td>
</tr>
<tr>
<td>images</td>
<td>/published-data/images/*</td>
<td>document-retrieval</td>
</tr>
<tr>
<td>images</td>
<td>/classification/cpc/media/*</td>
<td></td>
</tr>
</tbody>
</table>

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]
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.

```json
{
  "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"
    }
  ],
  "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. $1.1781815 \times 10^7$ 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/

![Figure 17: Unix time converted to Human readable time](image)

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

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:

<table>
<thead>
<tr>
<th>Endpoints</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulltext</td>
<td>Inquires about whether claims and/or descriptions are available for patents.</td>
</tr>
<tr>
<td>claims</td>
<td>Retrieves the <a href="#">patent claims</a>.</td>
</tr>
<tr>
<td>description</td>
<td>Retrieves the <a href="#">patent description</a>.</td>
</tr>
<tr>
<td>images</td>
<td>Inquiry and retrieval of images or &quot;drawings&quot;, which can also be PDF documents.</td>
</tr>
<tr>
<td>equivalents</td>
<td>Retrieves the <a href="#">simple patent family</a>; 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.)</td>
</tr>
<tr>
<td>biblio</td>
<td>The default response data for the published-data service, therefore the biblio endpoint is used for retrieval of and searches for <a href="#">bibliographic data</a>. This endpoint is actually to support requests for constituents (see next table for details.)</td>
</tr>
<tr>
<td>abstract</td>
<td>Retrieves the <a href="#">patent abstract</a> of the bibliographic data.</td>
</tr>
</tbody>
</table>

*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 you in responses.
<table>
<thead>
<tr>
<th>Constituents</th>
<th>Delivers</th>
</tr>
</thead>
<tbody>
<tr>
<td>full-cycle</td>
<td>the full publication cycle of the patent application. This consists of a &quot;full set&quot; or all of the publications related to an application.</td>
</tr>
<tr>
<td>images</td>
<td>drawings and descriptions of the patent (.png, .pdf, .tiff)</td>
</tr>
<tr>
<td>biblio</td>
<td>bibliographic data</td>
</tr>
<tr>
<td>abstract</td>
<td>patent abstract – This is only a constituent when the equivalents endpoint is used in the request (see Table 17: Endpoints above.)</td>
</tr>
</tbody>
</table>

Table 18

**Generic published-data request structure**

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

```plaintext
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:

<table>
<thead>
<tr>
<th>Services</th>
<th>Endpoints</th>
<th>Constituents</th>
<th>Can be combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Published-data services</td>
<td>/3</td>
<td>/biblio</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/abstract</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/full-cycle</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/image</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/fulltext</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/claims</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/description</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>/equivalents(^4)</td>
<td>/biblio</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/abstract</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/full-cycle</td>
<td>+</td>
</tr>
<tr>
<td>Bibliographic search</td>
<td>/4</td>
<td>/biblio</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/abstract</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/full-cycle</td>
<td>+</td>
</tr>
<tr>
<td>Family service</td>
<td>/4</td>
<td>/biblio</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/legal</td>
<td>+</td>
</tr>
<tr>
<td>Register retrieval</td>
<td>/3</td>
<td>/biblio</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/events</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/procedural-steps</td>
<td>+</td>
</tr>
<tr>
<td>Register search</td>
<td>/4</td>
<td>/biblio</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/events</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/procedural-steps</td>
<td>+</td>
</tr>
</tbody>
</table>

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:**

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Headers</th>
<th>Request Body</th>
</tr>
</thead>
</table>

**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.)
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:

```
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>
</bibliographic-data>

<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>
  </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>
  </bibliographic-data>
</exchange-document>

<!-- remaining bibliographic data stripped out for readability -->
</bibliographic-data>
For large numbers of input references we recommend using the **POST** method:

```

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:

```
Accept: application/exchange+xml

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 (8) for pressing and trimming clay in the mould containers, means (11) for supplying and placing tare-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>
Abstract

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

Request example:

Accept: application/exchange+xml

Accept: application/exchange+xml
Request Body: EP1000000.A1

Response:

![Figure 20](image-url)
Bibliographic search

The bibliographic search uses common query language (CQL) to retrieve the data. Additional query details are 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:

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
<?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>
            </publication-reference>
            <!-- classifications removed -->
            <application-reference doc-id="331447793">
              <document-id document-id-type="epodoc">
                <doc-number>KR20100130646</doc-number>
                <date>20101213</date>
              </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>
            </parties>
          </bibliographic-data>
        </exchange-document>
        <!-- more documents follow -->
      </exchange-documents>
    </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"/>
</ops:biblio-search>

Note, the maximum range is 100.

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 only available for the following authorities: EP, WO, AT, CA, CH, GB, FR and ES.

Request example:

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:

![Diagram of 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
<?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>
      <!-- Contents of the response -->
    </ops:publication-reference>
  </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.

![Diagram of fulltext retrieval](image)

---

**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:**

<table>
<thead>
<tr>
<th>Method</th>
<th>URI</th>
<th>Accept</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><a href="http://ops.epo.org/rest-services/published-data/publication/epodoc%5Cdescription">http://ops.epo.org/rest-services/published-data/publication/epodoc\description</a></td>
<td>application/fulltext+xml</td>
</tr>
</tbody>
</table>

Request Body: EP1000000

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

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:

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:

Accept: application/ops+xml

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

Response structure:

![Response structure diagram]

Figure 26
The structure shown in Figure 26 is part of the ops namespace and defined in the ops.xsd schema file.

Response example:

```xml
<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>
    </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:


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

**Retrieve the 1st drawing**

```
Accept: image/jpeg
```

![Figure 27: The result of "firstpage.jpeg" request](image)

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:

```
```
Accept: image/png

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:


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:


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:

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:

```
```

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.

```

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.

```
```

**Response:**

You get the requested page of the full document in PDF format for your request reference (EP1000000.A1):
3.1.4. Equivalents only or with abstract, biblio, full-cycle, images

This service retrieves the simple patent family data.

Request example:

```
Accept: application/ops+xml

POST http://ops.epo.org/rest-services/published-data/publication/epodoc/equivalents
Accept: application/ops+xml
Request Body: EP1000000
```
Response structure:

The structure shown in Figure 29 is part of the ops namespace and defined in the ops.xsd schema file.

Response example:

```xml
<?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:equivalents-inquiry>
  <ops:publication-reference>
    <document-id document-id-type="epodoc">
      <doc-number>NL1010536C</doc-number>
    </document-id>
  </ops:publication-reference>
</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:


Accept: application/ops+xml


Accept: application/ops+xml
Request Body: EP1000000

Response:

<?xml version="1.0" encoding="UTF-8"?>
  <ops:equivalents-inquiry>
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&gt;

Equivalents and biblio:

GET http://ops.epo.org/rest-services/published-

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

Response:

```xml
<?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>
            </publication-reference>
            <publication-reference>
              <document-id document-id-type="epodoc">
                <doc-number>NL1010536C</doc-number>
              </document-id>
            </publication-reference>
          </bibliographic-data>
        </exchange-document>
      </exchange-documents>
    </ops:inquiry-result>
  </ops:equivalents-inquiry>
</ops:world-patent-data>
```
Equivalents and bibliography and full-cycle:


Accept: application/ops+xml


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>
        </publication-reference>
        <publication-reference>
          <document-id document-id-type="epodoc">
            <doc-number>EP1000000</doc-number>
            <date>20000517</date>
          </document-id>
        </publication-reference>
      </bibliographic-data>
    </exchange-document>
  </exchange-documents>
</ops:inquiry-result>

<!-- remaining bibliographic data removed -->
</exchange-document>
</exchange-document>
</bibliographic-data>
</exchange-document>
</exchange-documents>
</ops:inquiry-result>
Equivalents and images:


Accept: application/exchange+xml


Accept: application/exchange+xml
Request Body: EP1000000.A1

Response:

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

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.

<table>
<thead>
<tr>
<th>Constituent</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>biblio</td>
<td>Delivers bibliographic data of each family member in the exchange-document element.</td>
</tr>
<tr>
<td>legal</td>
<td>Includes legal codes and descriptions of the family members.</td>
</tr>
</tbody>
</table>

Table 20

Generic request structure:

```plaintext
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:

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/ops+xml</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>POST</th>
<th><a href="http://ops.epo.org/rest-services/family/publication/docdb">http://ops.epo.org/rest-services/family/publication/docdb</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept</td>
<td>application/ops+xml</td>
</tr>
<tr>
<td>Request Body:</td>
<td><strong>EP.1000000.A1</strong></td>
</tr>
</tbody>
</table>

**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)


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>
            </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>
                </document-id>
            </priority-claim>
        </ops:family-member>
    </ops:patent-family>
</ops:world-patent-data>
3.2.2. Family data with biblio

Request:


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>
</ops:publication-reference>
</ops:family-member>
<ops:publication-reference>
<document-id document-id-type="docdb">
<country>AT</country>
<doc-number>232441T</doc-number>
<date>20030215</date>
</document-id>
</ops:publication-reference>
</ops:patent-family>
</ops:world-patent-data>
3.2.3. Family data with legal

Request:


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:

```xml
<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-active-indicator>YES</priority-active-indicator>
      </priority-claim>
      <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 20030815RER -CEASED AS TO PARAGRAPH 5 LIT. 3 LAW INTRODUCING PATENT TREATIES</ops:pre>
    </ops:legal>
    <ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
    <ops:L003EP desc="Document Number">99203729</ops:L003EP>
  </ops:family-member>
</ops:patent-family>
</ops:world-patent-data>
```
3.2.4. Family data with biblio and legal

Request:

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:family-member>
</ops:patent-family>
</ops:world-patent-data>
<ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
<ops:L003EP desc="Document Number">99203729</ops:L003EP>
<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: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>
</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:

![Figure 31](image-url)

Non-confidential © 2018 European Patent Office Page 89
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

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:

<table>
<thead>
<tr>
<th>Input format</th>
<th>Output format</th>
</tr>
</thead>
<tbody>
<tr>
<td>original</td>
<td>docdb, epodoc</td>
</tr>
<tr>
<td>docdb</td>
<td>epodoc, original</td>
</tr>
<tr>
<td>epodoc</td>
<td>original</td>
</tr>
</tbody>
</table>

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.

<table>
<thead>
<tr>
<th>Message Code</th>
<th>Message</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>pBRE001</td>
<td>Check digit not correct</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE002</td>
<td>No country code provided</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE003</td>
<td>Kind code not found</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE004</td>
<td>Date out of range</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE005</td>
<td>No number format detected for this kind code</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE006</td>
<td>Number transformation rule not specified</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE007</td>
<td>Number outside number range</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE008</td>
<td>Converted to Gregorian year</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE009</td>
<td>Converted to Emperor year</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE010</td>
<td>Date generated through lookup with input number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE011</td>
<td>No match found. Country code changed.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE012</td>
<td>Kind code in number does not match kind code supplied.</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE013</td>
<td>Output kind code different from extracted kind code.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE014</td>
<td>Kind code generated.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE015</td>
<td>Kind code transformed.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE016</td>
<td>Provided date is different from extracted date.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE017</td>
<td>Publication date set from extracted date.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE018</td>
<td>Application date set from extracted date.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE019</td>
<td>No kind code provided, setting it to A.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE020</td>
<td>Kind code does not start with a letter, setting it to A.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE021</td>
<td>Second character of provided kind code is not a digit, removing it..</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE022</td>
<td>Date is before year 1617</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE023</td>
<td>Date is after current year + 2</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE024</td>
<td>Month is not between 1 and 12, month and day set to 0</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE025</td>
<td>Day is invalid, set to 0</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE026</td>
<td>Date is in the future.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE027</td>
<td>Date lookup with input number did not find anything.</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE028</td>
<td>Date generated through lookup with output number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE029</td>
<td>Date lookup with output number did not find</td>
<td>ERROR</td>
</tr>
<tr>
<td>Code</td>
<td>Message</td>
<td>Level</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>pBRE030</td>
<td>Document with kind code found in master database.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE031</td>
<td>Kind code updated via lookup.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE032</td>
<td>Kind code lookup did not find anything.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE033</td>
<td>REFI kind code lookup did not find anything.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE034</td>
<td>REFI not allowed.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE035</td>
<td>Citing target set.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE036</td>
<td>Citing target not set, kind codes do not match.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE037</td>
<td>Date corrected via kind code lookup.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE038</td>
<td>Date set via kind code lookup.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE039</td>
<td>No Active Priority</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE040</td>
<td>Error in refi kind code lookup.</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE041</td>
<td>Last change origin of targetId was human, setting targetId as rawId.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE042</td>
<td>The provided document is a valid DOCDB format.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE043</td>
<td>The provided document is not a valid DOCDB format.</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE044</td>
<td>Rules without country used for transforming the number.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE045</td>
<td>Provided country code is different than extracted country code.</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE046</td>
<td>Kind code rule did not allow configured rules to transform number.</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE047</td>
<td>Meaningless digits removed from number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE048</td>
<td>Date generated through sequence number lookup</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE049</td>
<td>Date lookup with sequence did not find anything</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE050</td>
<td>RID found in master database</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE051</td>
<td>RID not found in master database</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE052</td>
<td>Document found by RID in master database</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE053</td>
<td>Document not found by RID in master database</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE054</td>
<td>Error in RID lookup</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE055</td>
<td>Document found in master database</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE056</td>
<td>Document not found in master database</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE057</td>
<td>preBRE not configured for validation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE058</td>
<td>Kind code is missing for validation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE059</td>
<td>Date is missing for validation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE060</td>
<td>Country code is missing for validation</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE061</td>
<td>Country code is invalid for validation</td>
<td>ERROR</td>
</tr>
<tr>
<td>pBRE100</td>
<td>Duplicate priority removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE101</td>
<td>Country code standardised</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE102</td>
<td>Leading/trailing characters removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE103</td>
<td>Country code removed for valid PCT number format</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE104</td>
<td>Chinese check digit removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE105</td>
<td>Country code derived from serial number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE106</td>
<td>Country code EP/EU changed to EM</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE107</td>
<td>Duplicate country code removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE108</td>
<td>Country code changed due to date</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE109</td>
<td>Country code derived from serial number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE110</td>
<td>Kind code set to W for PCT protection type</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE111</td>
<td>Kind code set to W for Epoque PCT priority</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE112</td>
<td>EP-A priority created from self-claimer</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE113</td>
<td>Kind code set to W for protection type PCT</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE114</td>
<td>LMI set for PCT priority</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE115</td>
<td>Country code set to WO for Epoque format</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE116</td>
<td>Country code removed for external PCT format</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE117</td>
<td>EP priority generated for HU translation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE118</td>
<td>LMI Removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE119</td>
<td>Kind code W forced to use the no country rules</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE120</td>
<td>Kind code set to W for application associated with publication country WO</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE121</td>
<td>Country code for priority starting with DM changed to XH</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE122</td>
<td>New country code set for KR citation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE123</td>
<td>Kind code derived from serial number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE124</td>
<td>Italian priority selected from multiple IT priorities provided</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE126</td>
<td>Kind code derived from number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE127</td>
<td>Kind code removed from number</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE128</td>
<td>JP single emperor year prepended with 0</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE129</td>
<td>Application kind code set to U for utility model publication</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE130</td>
<td>Application kind code set to T for national translation of EP</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE131</td>
<td>Spaces before/after separator removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE132</td>
<td>Leading zero added for single digit years (2000–2009)</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE133</td>
<td>Substitution of Chinese comma by a regular comma</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE134</td>
<td>New kind code U set for KR citation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE135</td>
<td>New raw ID set for KR citation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE136</td>
<td>New kind code set for KR citation</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE137</td>
<td>Transformed small caps characters to uppercase</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE138</td>
<td>Country code set to IB for DOCDB format</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE139</td>
<td>Priority removed</td>
<td>WARNING</td>
</tr>
<tr>
<td>pBRE999</td>
<td>Number could not be transformed, setting number from original</td>
<td>ERROR</td>
</tr>
</tbody>
</table>

Table 22
<table>
<thead>
<tr>
<th>Message Code</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>BRE000</td>
<td>Unknown error code</td>
</tr>
<tr>
<td>BRE001</td>
<td>Number Standardization Stopped</td>
</tr>
<tr>
<td>BRE002</td>
<td>Number Standardization Stopped – Routed to Post Processing</td>
</tr>
<tr>
<td>BRE003</td>
<td>RawId not Defined</td>
</tr>
<tr>
<td>BRE004</td>
<td>Country Code not Valid</td>
</tr>
<tr>
<td>BRE005</td>
<td>Exception in date lookup</td>
</tr>
<tr>
<td>BRE006</td>
<td>No Matching Pattern Found</td>
</tr>
<tr>
<td>BRE007</td>
<td>Failed to convert sequence number for document</td>
</tr>
<tr>
<td>BRE008</td>
<td>Illegal Country Code WO for APP-PRI</td>
</tr>
<tr>
<td>BRE009</td>
<td>Invalid Sequence Number in KindCodeSet</td>
</tr>
<tr>
<td>BRE010</td>
<td>Invalid Sequence Number in GenerateTargetId</td>
</tr>
<tr>
<td>BRE011</td>
<td>Invalid Length for SN</td>
</tr>
<tr>
<td>BRE012</td>
<td>SN Length Over Max Size</td>
</tr>
<tr>
<td>BRE013</td>
<td>Kind Code not valid for this Time Period</td>
</tr>
<tr>
<td>BRE014</td>
<td>Attempt to Apply Pattern that was not_parsed</td>
</tr>
<tr>
<td>BRE015</td>
<td>Failed to Convert Year Group</td>
</tr>
<tr>
<td>BRE016</td>
<td>Failed to Convert Sequence Number Group</td>
</tr>
<tr>
<td>BRW001</td>
<td>Country Code not Defined</td>
</tr>
<tr>
<td>BRW002</td>
<td>Document Date not Defined</td>
</tr>
<tr>
<td>BRW003</td>
<td>Kind Code not Defined</td>
</tr>
<tr>
<td>BRW004</td>
<td>Illegal Country Code for APP-PRI</td>
</tr>
<tr>
<td>BRW005</td>
<td>DM found in raw id, using XH country</td>
</tr>
<tr>
<td>BRW006</td>
<td>Using Default Country Pattern</td>
</tr>
<tr>
<td>BRW007</td>
<td>Country Code Replaced</td>
</tr>
<tr>
<td>BRW008</td>
<td>Year indication from date field</td>
</tr>
<tr>
<td>BRW009</td>
<td>Sequence number adjusted to</td>
</tr>
<tr>
<td>BRW010</td>
<td>Kind Code Relation not Allowed</td>
</tr>
<tr>
<td>BRW011</td>
<td>Failed to Define TargetId. RawId used instead</td>
</tr>
<tr>
<td>BRW012</td>
<td>Failed to Define TargetId. RawId used instead</td>
</tr>
<tr>
<td>BRW013</td>
<td>Failed to Define TargetId. RawId used instead</td>
</tr>
<tr>
<td>BRW014</td>
<td>No Kind Code or Generation Format Defined</td>
</tr>
<tr>
<td>BRW015</td>
<td>No Date Found. Trigger lookup</td>
</tr>
<tr>
<td>BRW016</td>
<td>Date Lookup Did not Find Anything</td>
</tr>
<tr>
<td>BRW017</td>
<td>Year Replacement Failed. No Date Defined in the Input</td>
</tr>
<tr>
<td>BRW018</td>
<td>Year Defined in Target Format, but no Date Defined</td>
</tr>
<tr>
<td>BRW019</td>
<td>Document Found in DB</td>
</tr>
<tr>
<td>BRW020</td>
<td>Kind code replaced by preferred lookup value</td>
</tr>
<tr>
<td>BRW021</td>
<td>Kind code replaced but not with first preferred lookup value</td>
</tr>
<tr>
<td>BRW022</td>
<td>Lookup found values which are not in preferred list</td>
</tr>
<tr>
<td>BRW023</td>
<td>Kind code lookup did not find anything</td>
</tr>
</tbody>
</table>
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:

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.infinity"?>
<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)

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
**service/application/docdb/original**

Accept: application/ops+xml
Request Body: **JP.2006147056.A.20060526**

Response:

```xml
<?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**

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
<?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
```

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
<?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

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/]
```

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:

<table>
<thead>
<tr>
<th>Constituent</th>
<th>Delivers</th>
</tr>
</thead>
<tbody>
<tr>
<td>biblio</td>
<td>bibliographic data</td>
</tr>
</tbody>
</table>
Table 24

<table>
<thead>
<tr>
<th>procedural-steps</th>
<th>the data fields that have occurred during the granting procedure of the requested application; only relevant for register service</th>
</tr>
</thead>
<tbody>
<tr>
<td>events</td>
<td>the information about actions that have taken place during the life of the patent application; only relevant for register service</td>
</tr>
</tbody>
</table>

**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:**


Accept: application/register+xml

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

Accept: application/register+xml

Request Body: EP99203729

**Response:**

```xml
<?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"
 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:document-id>
         </reg:publication-reference>
       </reg:bibliographic-data>
     </reg:register-document>
   </reg:register-documents>
 </ops:register-search>
</ops:world-patent-data>
```
Apparatus for manufacturing green bricks for the brick manufacturing industry.
Register data with events

Request:

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"?>
<ops:world-patent-data xmlns:ops="http://ops.epo.org"
xmns:reg="http://www.epo.org/register"
xmns: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>
Register data with procedural-steps

Request:


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"/>
</ops:register-search>
</ops:world-patent-data>
<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-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-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-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-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-data>
  </reg:register-document>
</reg:register-documents>
<reg:procedural-step procedure-step-phase="examination"
id="STEP_PROL_3">
</reg:procedural-step>
</reg:procedural-data>
</reg:register-document>
</reg:register-documents>
</ops:register-search>
</ops:world-patent-data>
Register data with biblio, events and procedural-steps

Request:

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">
<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>
</reg:bibliographic-data>
</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.

Register search support the same set of constituents and constituents combinations as a Register retrieval service.
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
<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="763">
    <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:application-reference>
            <reg:document-id>
              <reg:country>EP</reg:country>
            </reg:document-id>
          </reg:application-reference>
          <reg:classifications-ipcr>
            <reg:classification-ipcr sequence="CLASSIFICATION_1">H04N5/235</reg:classification-ipcr>
          </reg:classifications-ipcr>
        </reg:bibliographic-data>
      </reg:register-document>
    </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

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:

The structure shown in Figure 33 is part of the ops namespace and defined in the ops_legal.xsd schema file.

Request example:


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
<?xml version="1.0" encoding="UTF-8"?>
  <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>
        <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-active-indicator>YES</priority-active-indicator>
        </priority-claim>
        <ops:legal code="AK" desc="DESIGNATED CONTRACTING STATES:" infl="+"
          dateMigr="00010101">
          <ops:pre line="00001">EP 99203729A 2000-05-17AK</ops:pre>
        </ops:legal>
      </application-reference>
    </ops:family-member>
  </ops:patent-family>
</ops:world-patent-data>
```
CONTRACTING STATES: Kind Code of Ref Document A1

<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:L002EP desc="Filing / Published Document">F</ops:L002EP>
<ops:L003EP desc="Document Number">99203729</ops:L003EP>
<ops:L005EP desc="IPR Type">PI</ops:L005EP>
<ops:L006EP desc="PRS DOCUMENT TYPE">P</ops:L006EP>
<ops:L008EP desc="Legal Event Code 1">AK</ops:L008EP>
<ops:L500EP>
<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">EP 99203729A 2000-05-17AX +EXTENSION OF THE EUROPEAN PATENT TO Free Format Text AL;LT;LV;MK;RO;SI</ops:legal>

<ops:L002EP desc="Filing / Published Document">F</ops:L002EP>
<ops:L003EP desc="Document Number">99203729</ops:L003EP>
<ops:L005EP desc="IPR Type">PI</ops:L005EP>
<ops:L006EP desc="PRS DOCUMENT TYPE">P</ops:L006EP>
<ops:L008EP desc="Legal Event Code 1">AX</ops:L008EP>
<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

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:

The structure shown Figure 34 is part of the cpc namespace and defined in the CPCSchema.xsd schema file.

The classification response is \texttt{ops:classification-scheme}.

\textbf{3.6.1. CPC Retrieval}

\textbf{Note}, for more general information on the Cooperative Classification, please refer to: \url{http://www.cooperativepatentclassification.org}

Valid Query-string parameters in the CPC service

<table>
<thead>
<tr>
<th>Query-string</th>
<th>Description</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>class-title</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>notes-and-warnings</td>
<td></td>
</tr>
<tr>
<td>level</td>
<td></td>
</tr>
<tr>
<td>additional-only</td>
<td></td>
</tr>
<tr>
<td>sort-key</td>
<td></td>
</tr>
<tr>
<td>not-allocatable</td>
<td></td>
</tr>
<tr>
<td>breakdown-code</td>
<td></td>
</tr>
<tr>
<td>date-revised</td>
<td></td>
</tr>
<tr>
<td>link-file</td>
<td></td>
</tr>
<tr>
<td>depth</td>
<td>Determines how many children elements should be included in the response</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ancestors</td>
<td>Includes symbols above the requested element</td>
</tr>
<tr>
<td>navigation</td>
<td>Includes navigation symbols next &amp; previous in the response</td>
</tr>
</tbody>
</table>

**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"
```
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

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

Request for all children one level below classification A:
Accept: application/cpc+xml

POST http://ops.epo.org/rest-services/classification/cpc/?depth=1
Accept: application/cpc+xml
Request Body: A

Response:

```xml
<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: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>
        <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:class-title>
        </cpc:classification-item>
      </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:

<table>
<thead>
<tr>
<th>Format</th>
<th>Media type</th>
</tr>
</thead>
<tbody>
<tr>
<td>GIF</td>
<td>image/gif</td>
</tr>
<tr>
<td>JPEG</td>
<td>image/jpeg</td>
</tr>
<tr>
<td>TIF</td>
<td>image/tif</td>
</tr>
<tr>
<td>BMP</td>
<td>image/x-ms-bmp</td>
</tr>
<tr>
<td>PNG</td>
<td>image/png</td>
</tr>
<tr>
<td>WAV</td>
<td>audio/wav</td>
</tr>
<tr>
<td>WMA</td>
<td>audio/x-ms-wma</td>
</tr>
<tr>
<td>WMV</td>
<td>video/x-ms-wmv</td>
</tr>
<tr>
<td>MP3</td>
<td>audio/mpeg</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>type/x-unknown</td>
</tr>
</tbody>
</table>

Table 26
Example

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

```xml
<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:

```plaintext
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:**

![Diagram of the CPC search structure](image)

*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:

```xml
  <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: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:explanation>
              <cpc:text scheme="ipc">menstruation tables</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>
      <!-- 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

```plaintext
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 -->
</ops:classification-search>
```

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:

<table>
<thead>
<tr>
<th>Input format</th>
<th>Output format</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECLA</td>
<td>CPC</td>
</tr>
<tr>
<td></td>
<td>IPC</td>
</tr>
<tr>
<td>CPC</td>
<td>ECLA</td>
</tr>
<tr>
<td></td>
<td>IPC</td>
</tr>
</tbody>
</table>

Request structure

```plaintext
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>
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

4.1. Relevant acronyms and abbreviations

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

*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:
  o 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,
  o 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 singe CQL CPC classification index and can be combined with the other CQL operators and qualifiers within the same query.
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.

<table>
<thead>
<tr>
<th>Index name</th>
<th>Single correspondence?</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>title, ti</td>
<td>yes</td>
<td>the publication title in English</td>
</tr>
<tr>
<td>abstract, ab</td>
<td>yes</td>
<td>the publication abstract in English</td>
</tr>
<tr>
<td>titleandabstract, ta</td>
<td>no</td>
<td>the publication title or abstract in English</td>
</tr>
<tr>
<td>inventor, in</td>
<td>yes</td>
<td>an inventor name</td>
</tr>
<tr>
<td>applicant, pa</td>
<td>yes</td>
<td>an applicant name</td>
</tr>
<tr>
<td>inventorandapplicant, ia</td>
<td>no</td>
<td>a name of an inventor or an applicant</td>
</tr>
<tr>
<td>publicationnumber, pn</td>
<td>yes</td>
<td>the publication number in any format</td>
</tr>
<tr>
<td>spn</td>
<td>yes</td>
<td>the publication number in epodoc format</td>
</tr>
<tr>
<td>applicantnumber, ap</td>
<td>yes</td>
<td>the application number in any format</td>
</tr>
<tr>
<td>sap</td>
<td>yes</td>
<td>the application number in epodoc format</td>
</tr>
<tr>
<td>prioritynumber, pr</td>
<td>yes</td>
<td>the priority number</td>
</tr>
<tr>
<td>spr</td>
<td>yes</td>
<td>the priority number in epodoc format</td>
</tr>
<tr>
<td>num</td>
<td>no</td>
<td>the publication, application or a priority number in any format</td>
</tr>
<tr>
<td>publicationdate, pd</td>
<td>yes</td>
<td>the publication date in one of the following formats:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• YYYY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• yyyyMM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• yyyyMMdd</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• yyyy-MM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• yyyy-MM-dd</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MM/yyyy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• dd/MM/yyyy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MM.yyyy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• dd.MM.yyyy</td>
</tr>
<tr>
<td>citation, ct</td>
<td>yes</td>
<td>a cited document number</td>
</tr>
<tr>
<td>ex</td>
<td>yes</td>
<td>a cited document number during the examination</td>
</tr>
<tr>
<td>op</td>
<td>yes</td>
<td>a cited document number during the opposition</td>
</tr>
<tr>
<td>Index name</td>
<td>Single correspondence?</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>rf</td>
<td>yes</td>
<td>a cited document number provided by the applicant</td>
</tr>
<tr>
<td>oc</td>
<td>yes</td>
<td>another cited document number</td>
</tr>
<tr>
<td>famn</td>
<td>yes</td>
<td>a simple family identifier</td>
</tr>
<tr>
<td>cpc</td>
<td>yes</td>
<td>an CPC classification</td>
</tr>
<tr>
<td>cpci</td>
<td>yes</td>
<td>CPC invention classifications</td>
</tr>
<tr>
<td>cpca</td>
<td>yes</td>
<td>CPC additional classifications</td>
</tr>
<tr>
<td>cpcc</td>
<td>yes</td>
<td>CPC linked symbols confirmed + National Office</td>
</tr>
<tr>
<td>ipc, ic</td>
<td>no</td>
<td>any IPC1-8 class</td>
</tr>
<tr>
<td>ci</td>
<td>yes</td>
<td>IPC8 core invention class</td>
</tr>
<tr>
<td>cn</td>
<td>yes</td>
<td>IPC8 core additional class (non-invention)</td>
</tr>
<tr>
<td>ai</td>
<td>yes</td>
<td>IPC8 advanced invention class</td>
</tr>
<tr>
<td>an</td>
<td>yes</td>
<td>IPC8 advanced additional class (non-invention)</td>
</tr>
<tr>
<td>a</td>
<td>no</td>
<td>any IPC8 advanced class</td>
</tr>
<tr>
<td>c</td>
<td>no</td>
<td>any IPC8 core class</td>
</tr>
<tr>
<td>cl</td>
<td>no</td>
<td>an CPC or IPC1-8 class</td>
</tr>
<tr>
<td>txt</td>
<td>no</td>
<td>publication title or abstract, or inventor/applicant name</td>
</tr>
</tbody>
</table>

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 &lt; (including the semicolon) and the 'greater than' symbol >, must become &gt; (including the semicolon).

<table>
<thead>
<tr>
<th>Query</th>
<th>Description of result</th>
</tr>
</thead>
<tbody>
<tr>
<td>ti all &quot;green energy&quot;</td>
<td>All publications where &quot;green&quot; and &quot;energy&quot; are found among all titles. <em>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.</em></td>
</tr>
<tr>
<td>ti=&quot;green energy technology&quot;</td>
<td>All publications where a title contains exact words/phrase &quot;green energy technology&quot;</td>
</tr>
<tr>
<td>ti=green prox/unit= paragraph ti=energy</td>
<td>All publications where a title contains &quot;green&quot; and &quot;energy&quot; together in a same paragraph.</td>
</tr>
<tr>
<td>pd within &quot;20051212 20051214&quot;</td>
<td>All publications published 12, 13 or 14 December 2005</td>
</tr>
<tr>
<td>pd=&quot;20051212 20051214&quot;</td>
<td>Same as above</td>
</tr>
<tr>
<td>ia any &quot;John Smith&quot;</td>
<td>All publications with &quot;John&quot; and/or &quot;Smith&quot; in the applicant/inventor name</td>
</tr>
<tr>
<td>pn=EP and pr=GB</td>
<td>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)</td>
</tr>
<tr>
<td>ta=green prox/distance&lt;=3 ta=energy</td>
<td>All publications having words &quot;green&quot; and &quot;energy&quot; in either title or abstract separated by at most 3 words from each other,</td>
</tr>
<tr>
<td>ta=green prox/distance&lt;=2/ordered=true ta=energy</td>
<td>All publications having a word &quot;green&quot; followed by a word &quot;energy&quot; in either title or abstract separated by at most 2 words,</td>
</tr>
<tr>
<td>(ta=green prox/distance&lt;=3 ta=energy) or (ta=renewable prox/distance&lt;=3 ta= energy)</td>
<td>All publications having either &quot;green&quot; or &quot;renewable&quot; separated by at most 3 words from a word &quot;energy&quot; in either title or abstract, in any order.</td>
</tr>
<tr>
<td>pa all &quot;intelligence agency atomic&quot; and JP</td>
<td>All publications with a publication/application/priority document issued by Japan (having a JP country code in a document number) having &quot;intelligence&quot;, &quot;agency&quot;, &quot;atomic&quot; words in an applicant name</td>
</tr>
<tr>
<td>Query</td>
<td>Description of result</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>pa all &quot;intelligence agency atomic&quot; and JP and pd&gt;2000</td>
<td>See above, but published after 2000.</td>
</tr>
<tr>
<td>pd &lt; 18000101</td>
<td>All publications published before 18th century.</td>
</tr>
<tr>
<td>ta=synchroni#ed</td>
<td>Allows to look for &quot;synchronized&quot; or &quot;synchronised&quot; word in abstract/title</td>
</tr>
<tr>
<td>EP and 2009 and Smith</td>
<td>All publications with application/publication/priority document issued by EPO (having EP country code), published in 2009, having Smith as an applicant or inventor</td>
</tr>
<tr>
<td>cpc=/low A01B</td>
<td>All publications classified with A01B and all subclasses</td>
</tr>
<tr>
<td>ct=EP1027777</td>
<td>All publications citing EP1027777</td>
</tr>
<tr>
<td>txt all &quot;vision technologies&quot;</td>
<td>All publications where words “vision” and “technology” can be found either in title, abstract, inventor or applicant name</td>
</tr>
</tbody>
</table>

*Table 30*
Extended CQL examples

OPS also supports so-called extended CQL language (see “Smart Search”).

<table>
<thead>
<tr>
<th>Input Query</th>
<th>Standard CQL equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>G08B25 H04L63</td>
<td>cl=G08B25 and cl=H04L63</td>
</tr>
<tr>
<td>G08B25 H04L63 title=grid</td>
<td>cl=G08B25 and cl=H04L63 and title=grid</td>
</tr>
<tr>
<td>Siemens EP 200701</td>
<td>inventorandapplicant=Siemens and num=EP and publicationdate=200701</td>
</tr>
<tr>
<td>20070101:20070115</td>
<td>publicationdate within &quot;20070101 20070115&quot;</td>
</tr>
</tbody>
</table>

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:

```xml
<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)
JP30000014B (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)
4.4. Expressing PCT application numbers in docdb format

The format has changed through the years:
- Until 1.1.2004: CCyyynnnnW
- After 1.1.2004: CCccyyynnnnnnW

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

Accept: application/ops+xml

Response:

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 \url{http://ops.epo.org/rest-services/family/priority/docdb/US.18314305.A}

Accept: application/ops+xml

Response:

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:

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

![Figure 40](image)

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

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

Response:

![Figure 41](image)

**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/

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