Europäisches Patentamt
European Patent Office
Office européen des brevets

FEDERATED EUROPEAN PATENT REGISTER SERVICE

Manual of technical requirements and guidelines

Version 0.5 - September 2014

Summary

This document is for the national patent offices to set up a RESTful web service using JSON.

LEGAL NOTICE

Trade marks and logo

The EPO official logo is likewise protected worldwide as an emblem of an international organisation under the Paris Convention for the Protection of Industrial Property.

Disclaimer

The EPO provides this document and adaptable XML schema to various national patent offices (NO) only as a guide to independently setting up the Federated European Patent Register (search and retrieval) Service. The EPO accepts no liability for adaptations to the XSD provided, for the web services or for database configurations by any NO to implement this service.

Other access conditions

The EPO reserves the right to modify, extend or discontinue the available services, in full or in part, without prior notice.

DOCUMENT INFORMATION

Contact for inquiries:	
Person	Stephane Rey-Coquais
Email	sreycoquais@epo.org

Document history			
Document version	**Date**	**Changes**	**Authors**
0.1	MAY 2013	New document	Patricia Passarelli (technical editor) Eugeny Burdo (developer)
0.2	MAY 2013	Replaced HTTP with HTTPS	Eugeny Burdo (developer)
0.3	JULY 2013	Added NO Status list	Stephane Rey-Coquais (administrator) Patricia Passarelli (technical editor)
0.4	OCTOBER 2013	The text length is increased (150 characters instead of 35)	Eugeny Burdo (developer)
0.5	SEPTEMBER 2014	Change business logic for invalidation date (see *§2.5.2.*	Eugeny Burdo (developer)

		Data from the NOs)	

TABLE OF CONTENTS

1. INTRODUCTION

The Federated European Patent Register service (short name 'Federated register') has been built into the European Patent (EP) Register as an additional functionality. When EP Register users view an EP document, they have the option to view the main legal information of that document in the national phase, via the Federated register which is a panel right inside the EP Register user interface.

Thus, the EPO invites the national offices (NO) to build a RESTful web service to handle data retrieval via "Asynchronous JavaScript + XML" (AJAX) requests from client web browsers visiting the EP Register and deliver "JSON with padding" (JSONP). This JSONP must be based on XML, which must be validated against the XSD provided by the EPO. This data will then be displayed in a single, unified table.

1.1. AUDIENCE AND PURPOSE

This guide is for system developers or software engineers who will use the requirements and guidelines contained herein to:

- set up a RESTful web service, that
- provides JSONP output, that
- is derived from XML, that
- complies with the XSD (created by the EPO and provided with this guide.)

Thus, it is anticipated that these developers or software engineers will use their knowledge of databases, RESTful technologies, JSON(P), XML, and XSD to deliver NO legal status data to the client web browser querying the EP Register.

This guide also provides information on the overview of data items requested for the table view (see *§2.5. Data items displayed*). Therefore a business specialist at the NO will need to use the necessary input details (see *§2.5.2. Data from the NOs*) to match the request to the local data interpretation.

Technical support for setting up a RESTful web service is provided by the EPO:

- in the form of this guide,
- online documents available on Epoxy (technical portal for NOs),
- through knowledge transfer with EPO experts (ad-hoc basis).

1.2. DATA STANDARDS AT THE EPO

JSONP is one of the standards used at the EPO to transmit data over the Internet, e.g. via the EP Register web service.

The RESTful web service built by the NOs must be able to deliver JSONP. To produce JSONP, it must be translated from the initial XML, which complies with the XSD provided by the EPO. Furthermore, the JSON must be compliant with the BadgerFish convention (see *§3.3.3. Data validation*.)

1.3. PLANNING AND TIMELINE

At the end of March 2013 the EPO implemented the Federated register table view within the EP Register. This implementation is ready to receive NO input.

For each NO that indicates to the EPO that they are ready with the RESTful web service for data provision, a test cycle is started for the technical and business validation of the input provided.

Once the NO and EPO agree to be ready to deliver the data for the use by the general public, the EPO will set up the EP Register to allow the display of NO data within the Federated register. This update is done on the fly; it does not require a new release and needs approximately one hour to take effect.

It is the intention of the EPO to extend the Federated register with new features, which will always be announced in the EP Register editorial and release notes, in the Patent Information (PI) news as well as at upcoming PI conferences and workshops.

2. THE FEDERATED REGISTER

The Federated register aims to improve access to national office (NO) legal data and provide a comprehensive overview of information from multiple EP member state NO patent registers.

2.1. ACCESS

A navigation tab to the Federated register panel will appear in the file navigation menu, placed below '*Legal status*'. The Federated register panel will be accessible on all panel views of the EP Register:

Figure: *access to the Federated Register*

2.2. GRAPHICAL USER INTERFACE (GUI)

The Federated register will display legal status data in a separate panel inside the EP Register GUI.

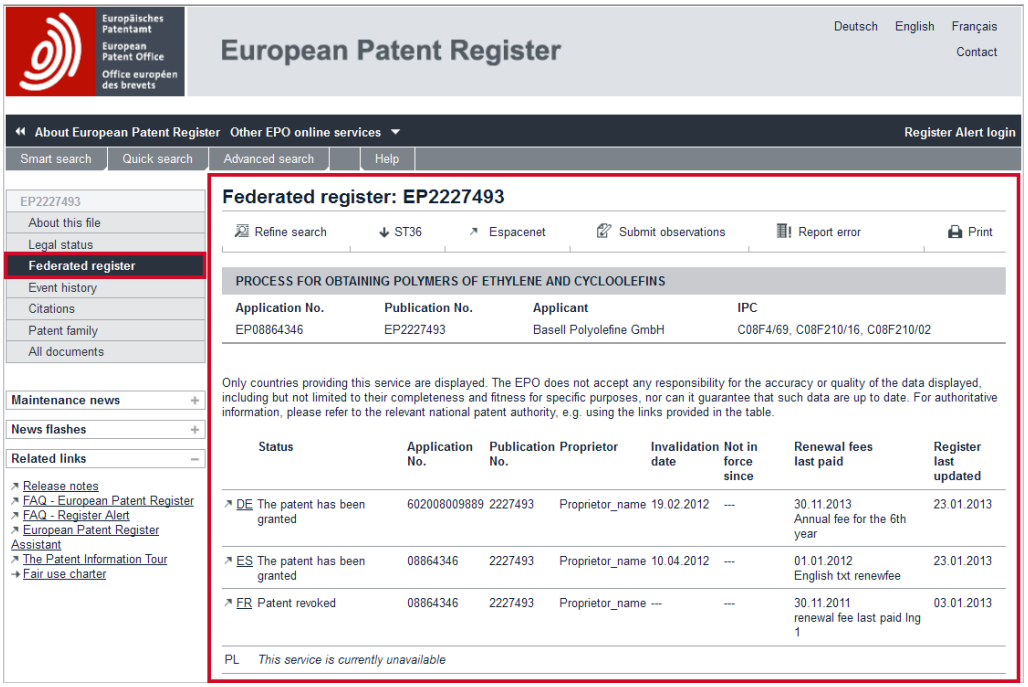


Figure: *EP Register GUI with Federated Register*

2.3. REQUEST AND RECEIVE DATA

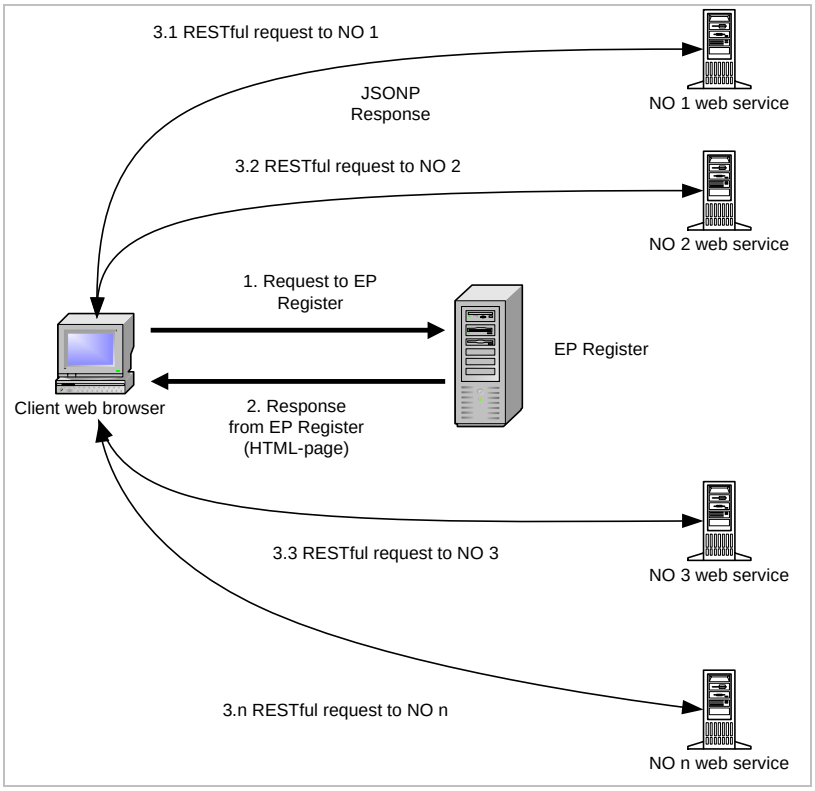


Figure: *request and receive data*

2.3.1. Scenario

The client (a web browser) sends a request to the EP Register. The request is processed and HTML is returned to the client as a response.

When the HTML is displayed in the client, the client sends AJAX requests to the NOs (in parallel) to retrieve legal status data.

When a response from the NO is received it will be processed on the client side and the data will be displayed to the users (as shown in §2.2. Graphical user interface.)

2.4. PROCESS RECEIVED DATA

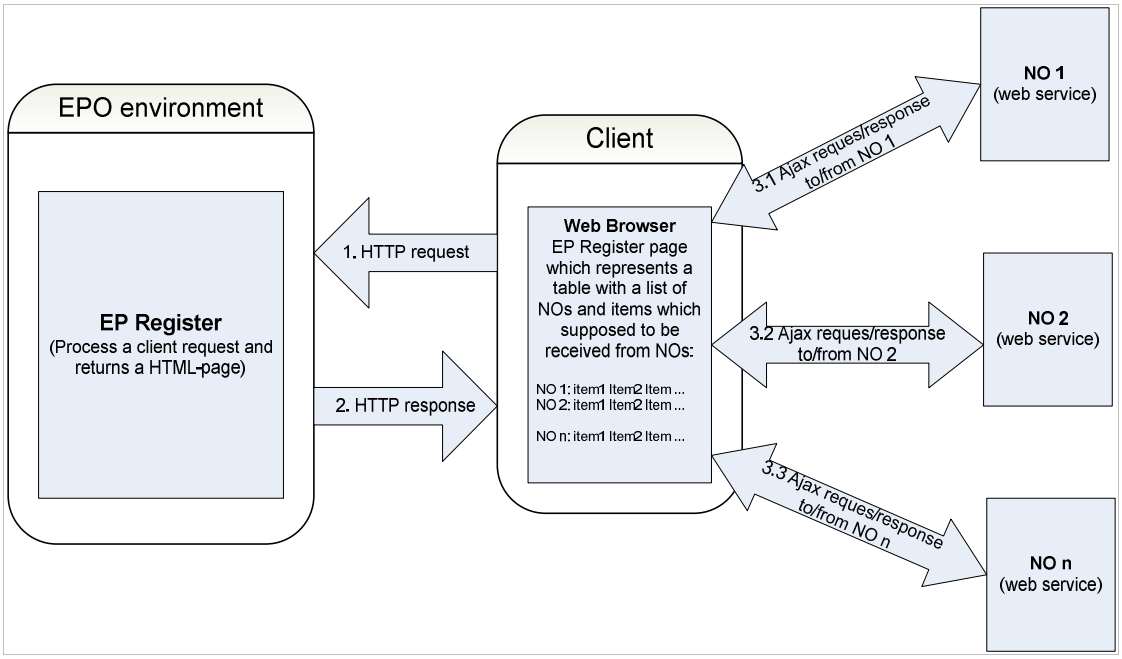


Figure: *data process and flow*

2.4.1. Data flow

1. Client: a user interacts with the EP Register via their web browser and sends a request for 'federated' data (Federated register panel) to the EPO environment (EP Register);

2. EPO environment: based on the requested data the EP Register responds to the client with a dynamically built HTML page. This HTML page contains a 'federated' table containing the built AJAX requests ready to be issued to national offices in parallel;

3. Client (Federated register GUI): the HTML page will be displayed in the browser and AJAX requests will be issued (in parallel) to NO's;

4. NO 1, NO 2, NO 3, ...: national offices will process the requests and each NO (asynchronously) responds with JSONP;

5. Client (Federated register GUI): When the client receives the data, it will process and display it in the Federated register. When the received data is not valid for a specific office or an office does not respond within five seconds (maximum) an error message referring to that NO will be shown in the GUI. (See §2.9. *Service unavailable*.)

2.4.2. HTTPS connection

The service should provide the HTTPS connection. This secured connection will be used by the EP Register to retrieve legal data.

Thereafter the "HTTP" abbreviation will be used in the document (HTTP**S** is just a secured version of HTTP).

2.4.3. Endpoint structure

It is up to each individual NO to provide the endpoint. The required URL should at least include a number and the specification of whether this number is an application or a publication number.

For instance, the URL for an application number should be:

```
https://....../AP/CCNNNNNNNN.js?callback=FunctionName
```

The URL for a publication number should be:

```
https://....../PN/CCNNNNNNN.js?callback=FunctionName
```

Structure parts	Description	Mandatory
`.....`	**Decided by each individual NO**	Yes
AP	application number	Yes; should be either one of these two
PN	publication number	
CC	Country code (for the EPO it's 'EP')	Yes
NNNNNNNN	Application (8 digits) number	Yes; should be either one of these two
NNNNNNN	Publication (7 digits) number	
`.js`	request format (see §3.3.4 *JSONP technology*)	Yes
`?callback=FunctionName`	Requested function name; it should default to "`parseResponse`"	No; if not provided the return function should default to "`parseResponse`"

Table: *endpoint structure*

2.4.3.1. Endpoint example

A request for **application or publication number** looks like this:

`https://...../AP/EP00101010.js?callback=jQuery1910508136190939`

`https://...../PN/EP2227493.js?callback=jQuery1910508136190939`

2.5. DATA ITEMS DISPLAYED

A maximum of eight (8) different data items *per document, per country* will be displayed. In the future the amount of data items may increase depending on feedback.

	Status	Application No.	Publication No.	Proprietor	Invalidation date	Not in force since	Renewal fees last paid	Register last updated
DE	The patent has been granted	602008009889	2227493	Proprietor_name	19.02.2012	---	30.11.2013 Annual fee for the 6th year	23.01.2013
ES	The patent has been granted	08864346	2227493	Proprietor_name	10.04.2012	---	01.01.2012 English txt renewfee	23.01.2013
FR	Patent revoked	08864346	2227493	Proprietor_name	---	---	30.11.2011 renewal fee last paid lng 1	03.01.2013

Figure: *data items displayed*

2.5.1. Data from the EP Register

Data item	Header Titles
`title (text)`	----
`EP application Number`	Application No.
`EP publication Number`	Publication No.
`EP applicant name (text)`	Applicant
`IPC`	IPC

Table: *data from EP Register*

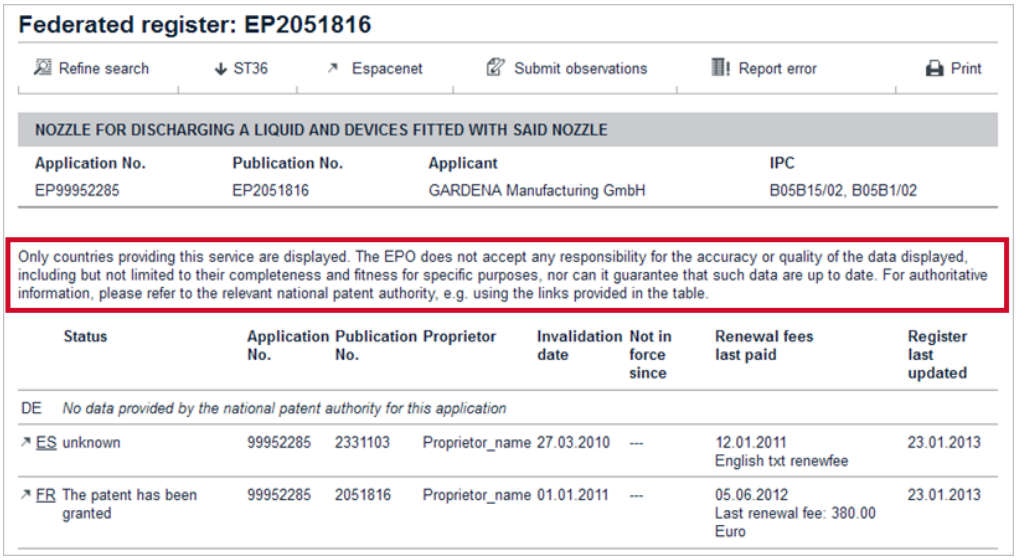
2.5.2. Data from the NOs

Data item	Column Headers
`application number (number)`	Application No.
`publication number (number)`	Publication No.
`proprietor name (text)`	Proprietor
`status (code)`	**Status** (Please refer to § 2.12. *Status*)
`patent withdrawn (date)`	**Invalidation date:** Only one of these dates will ever be displayed. if more than one date is provided, the most recent one will be shown in the Federated table
`patent revoked (date)`	
`patent lapsed (date)`	
`patent expired (date)`	
`not in force (date)`	Not in force since
`renewal fees last paid (date)`	Renewal fees last paid
`last update of the nat. register (date)`	Register last updated

Table: *data from NOs*

2.5.3. Disclaimer

The EP Register disclaims to its users that the accuracy of data displayed in the Federated register is derived from the XML/JSONP provided by each NO. Thus each NO is responsible for the quality of data made available to the application.



Federated register: EP2051816

Refine search | ST36 | Espacenet | Submit observations | Report error | Print

NOZZLE FOR DISCHARGING A LIQUID AND DEVICES FITTED WITH SAID NOZZLE

Application No.	Publication No.	Applicant	IPC
EP99952285	EP2051816	GARDENA Manufacturing GmbH	B05B15/02, B05B1/02

Only countries providing this service are displayed. The EPO does not accept any responsibility for the accuracy or quality of the data displayed, including but not limited to their completeness and fitness for specific purposes, nor can it guarantee that such data are up to date. For authoritative information, please refer to the relevant national patent authority, e.g. using the links provided in the table.

	Status	Application No.	Publication No.	Proprietor	Invalidation date	Not in force since	Renewal fees last paid	Register last updated
DE	*No data provided by the national patent authority for this application*							
ES	unknown	99952285	2331103	Proprietor_name	27.03.2010	---	12.01.2011 English txt renewfee	23.01.2013
FR	The patent has been granted	99952285	2051816	Proprietor_name	01.01.2011	---	05.06.2012 Last renewal fee: 380.00 Euro	23.01.2013

Figure: *data disclaimer*

2.5.4. Required data formatting (from the NOs)

Any text provided in the data response will be displayed. However, each NO should aim to deliver data like this:

- **date:** ISO8601 short format: YYYYMMDD (per the XSD)
- **text:** a maximum of 150 characters is allowed by the XSD. The first 35 characters of the text will be displayed in the Federated table. The whole text (i.e. 150 characters) will be available via the tooltip.
- **status:** (see key values in *§2.12. Status*.)

Please note, number formatting can vary but please consult the XSD to envisage output in the Federated Register GUI.

2.5.5. Language priority

The EPO would prefer to receive NO data in the official language used in the EP Register: English, French, and German. These are the languages

that users are familiar with already. However, if an NO provides data in its national language that is not English, French or German, that language will be displayed according to this set of priorities:

EP Register language interfaces		EN	DE	FR
Language priority	**1**	EN	DE	FR
	2	DE	EN	EN
	3	FR	FR	DE
	4	national language	national language	national language
	5	any other language	any other language	any other language

Table: *language display priority*

2.5.6. Empty fields

It is possible that a field will be empty.

When the JSONP retrieved from the NO side does not contain a tag with the specified key (see *§3.4. RESTful service example*), the data item will be displayed with dashes (see figure below).
When users run their mouse over these dashes they will see a message:

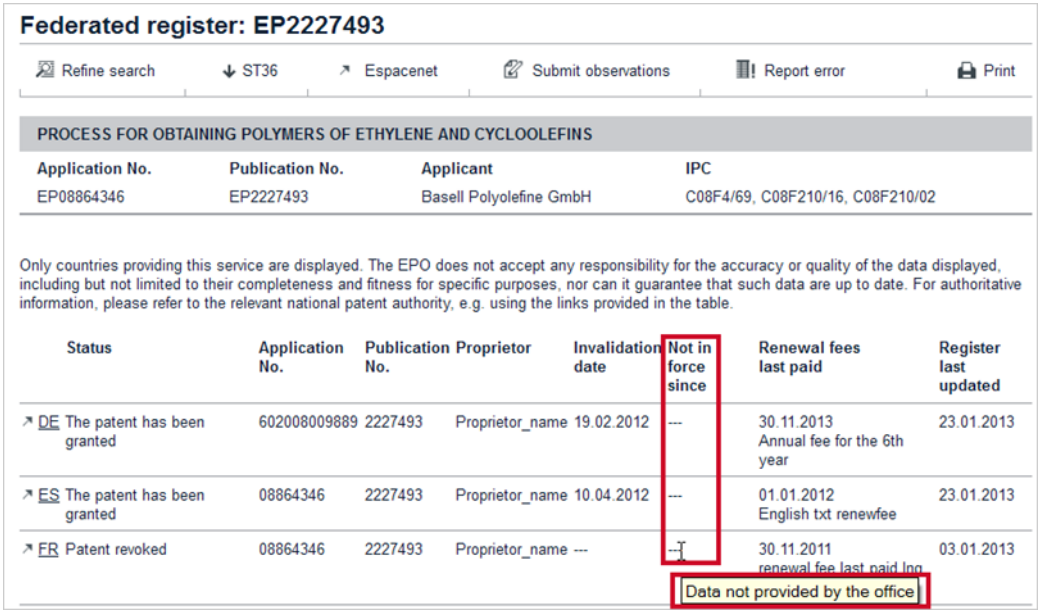


Figure: *empty fields' message*

2.6. ERROR MESSAGES

Error messages are based on the standard HTTP status codes. A link for these codes is listed in *§4. Useful links*.

2.7. HYPERLINKS TO NATIONAL PATENT REGISTERS

Hyperlinks are built from the JSONP input received from each NO. These hyperlinks will directly lead EP Register users to their chosen patent document within the respective NO patent register.

When an NO does not provide user access to its patent data, this hyperlink will not be created nor will it be available to EP Register users.

This will be improved in the future: the EP Register will always provide a hyperlink regardless of whether an NO provides access to its patent data.

In the Federated register panel a 'tooltip' message is displayed for each NO hyperlink to inform users they can click on it to reach that particular national patent register:

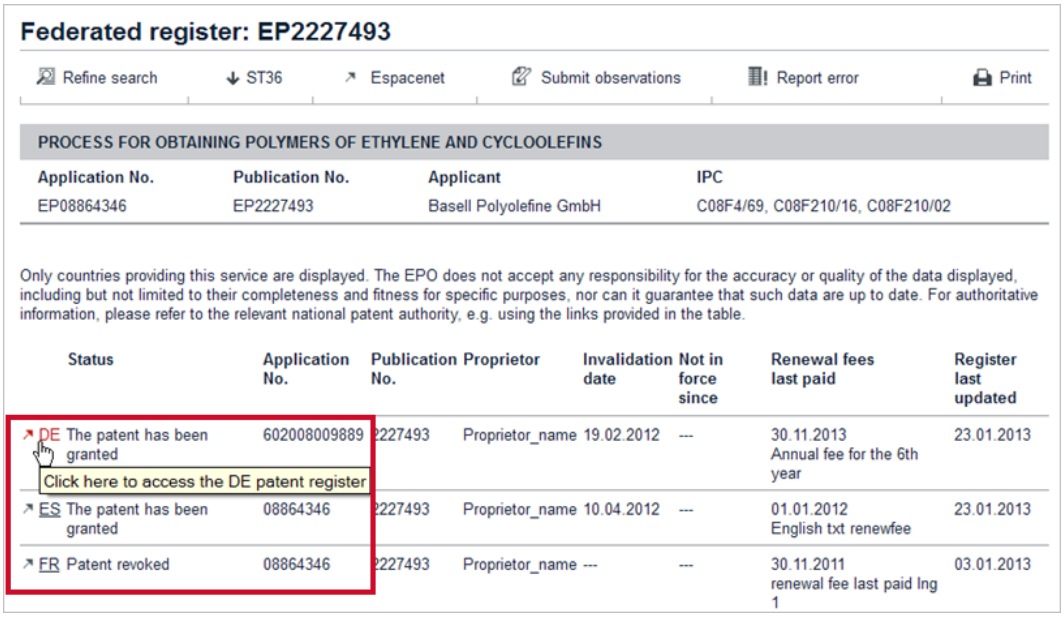


Figure: *hyperlinks to NOs*

2.8. LOADING

The response time from NO web services should be < 5 seconds (ultimately it should be ~3 seconds.) Loading occurs as data becomes available (asynchronously):

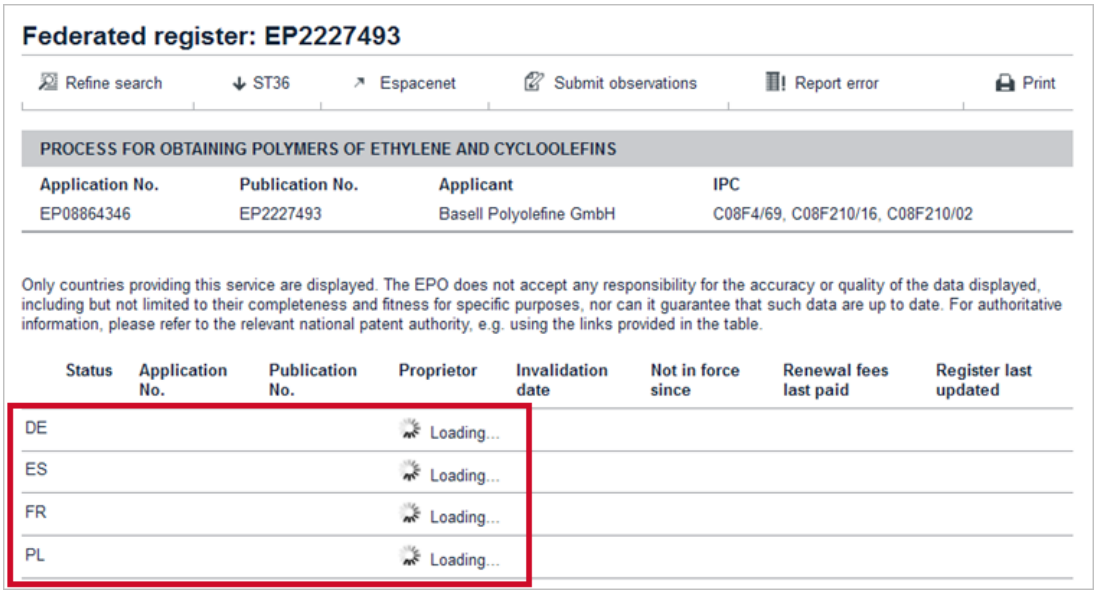


Figure: *loading*

2.9. SERVICE UNAVAILABLE

In cases when the response time exceeds the maximum limit of five (5) seconds, it is presumed that the service is currently unavailable. In such cases a message will be shown to the user:

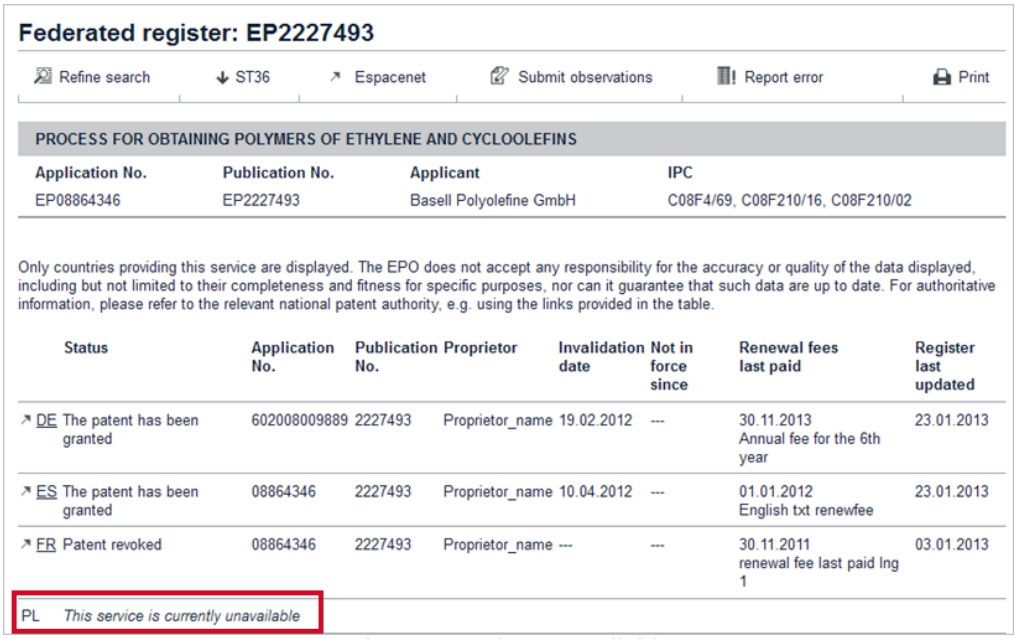


Figure: *service unavailable*

2.10. UNAVAILABLE DATA

There could be instances when data from an NO is not retrievable or the NO web service is temporarily unavailable. There could also be an instance in which an NO provides the RESTful service but does not or cannot supply data for a particular patent application (see figure below).

In either case the user will see a message:

Federated register: EP2051816

Refine search | ST36 | Espacenet | Submit observations | Report error | Print

NOZZLE FOR DISCHARGING A LIQUID AND DEVICES FITTED WITH SAID NOZZLE

Application No.	Publication No.	Applicant	IPC
EP99952285	EP2051816	GARDENA Manufacturing GmbH	B05B15/02, B05B1/02

Only countries providing this service are displayed. The EPO does not accept any responsibility for the accuracy or quality of the data displayed, including but not limited to their completeness and fitness for specific purposes, nor can it guarantee that such data are up to date. For authoritative information, please refer to the relevant national patent authority, e.g. using the links provided in the table.

Status	Application No.	Publication No.	Proprietor	Invalidation date	Not in force since	Renewal fees last paid	Register last updated
DE *No data provided by the national patent authority for this application*							
ES unknown	99952285	2331103	Proprietor_name	27.03.2010	---	12.01.2011 English txt renewfee	23.01.2013
FR The patent has been granted	99952285	2051816	Proprietor_name	01.01.2011	---	05.06.2012 Last renewal fee: 380.00 Euro	23.01.2013

Figure: *unavailable data*

2.11. PERFORMANCE

By providing the Federated register to EP Register users, the NOs should expect a spike in traffic. As an indication, the EP Register legal data page receives approximately 35.000 requests per day.

NOs must independently manage traffic spikes and overloads. In addition the NOs are independently responsible for access risks such as hack attempts.

2.12. STATUS

The XSD does not describe any of the status keys. Status should be retrieved from the <code-value> tag with the key "Patent Status" included in the XML. Status keys must come from the common list (see table below) which corresponds to the patent status list built by consolidating information provided by National Offices (NO's).

Status number	Patent status : EN	Patent status : FR	Patent status: DE
NO_STATUS_1	Application deemed to be withdrawn	Demande réputée retirée	Anmeldung gilt als zurückgenommen
NO_STATUS_2	Application withdrawn	Demande retirée	Anmeldung zurückgenommen
NO_STATUS_3	Application published	Demande publiée	Anmeldung veröffentlicht
NO_STATUS_4	Application refused	Demande rejetée	Anmeldung zurückgewiesen
NO_STATUS_5	Formalities in progress	Examen quant à la forme en cours	Formalprüfung läuft
NO_STATUS_6	Examination in progress	Examen en cours	Prüfung läuft
NO_STATUS_7	Examination requested	Examen demandé	Prüfung beantragt
NO_STATUS_8	Grant of patent intended	Délivrance d'un brevet projetée	Patenterteilung beabsichtigt
NO_STATUS_9	Granted	Délivré	Erteilt
NO_STATUS_10	International application published	Demande internationale publiée	Internationale Anmeldung veröffentlicht
NO_STATUS_11	No opposition filed within time limit	Pas d'opposition formée dans les délais	Kein fristgerechter Einspruch eingelegt
NO_STATUS_12	Opposition procedure closed	Procédure d'opposition terminée	Einspruchsverfahren abgeschlossen
NO_STATUS_13	Opposition pending	Opposition en instance	Einspruch anhängig
NO_STATUS_14	Opposition rejected	Opposition rejetée	Einspruch zurückgewiesen
NO_STATUS_15	Request for EP validation received	Requête en validation EP reçue	EP-Validierungsantrag eingegangen
NO_STATUS_16	Patent limited	Brevet limité	Patent beschränkt
NO_STATUS_17	Patent renounced	Brevet auquel il est renoncé	Verzicht auf das Patent erklärt
NO_STATUS_18	Patent maintained in amended form	Brevet maintenu sous une forme modifiée	Patent in geänderter Fassung aufrechterhalten
NO_STATUS_19	Patent revoked	Brevet révoqué	Patent widerrufen
NO_STATUS_20	Patent revoked by applicant	Brevet révoqué par le demandeur	Patent vom Anmelder widerrufen
NO_STATUS_21	Patent lapsed	Brevet éteint	Patent erloschen
NO_STATUS_22	Patent expired	Brevet expiré	Patent abgelaufen

NO_STATUS_23	Patent surrendered	Brevet auquel il est renoncé	Verzicht auf das Patent erklärt
NO_STATUS_24	Patent secret	Brevet confidentiel	Geheimpatent
NO_STATUS_25	Patent validated	Brevet validé	Patent validiert
NO_STATUS_26	Patent not validated	Brevet non validé	Patent nicht validiert
NO_STATUS_27	Patent in force	Brevet en vigueur	Patent in Kraft
NO_STATUS_28	Patent not in force	Brevet non en vigueur	Patent nicht in Kraft
NO_STATUS_29	Appeal filed	Recours formé	Beschwerde eingelegt
NO_STATUS_30	Nullity requested	Nullité demandée	Nichtigkeit beantragt
NO_STATUS_31	SPC granted	CCP délivré	ESZ erteilt
NO_STATUS_32	Unknown	Inconnu	Unbekannt

Table: *Status list as of 01-07-2013*

For example the following data will allow the display of the **status** "***Application withdrawn***" (for the patent concerned):

```
<code-value key="Patent Status">
  <code>NO_STATUS_2</code>
</code-value>
```

Here is the **example** in **JSON:**

```
{
            "code-value":{
             "code":{
               "$":"NO_STATUS_2"
                 },
                 "@key":"Patent Status"
            },
        }
```

3. RESTFUL WEB SERVICE

3.1. DEFINITION

REST refers to **RE**presentational **S**tate **T**ransfer, which is a predominant web service design model. A web service built with REST is often known as a RESTful service.

The RESTful service provided by each NO should follow these basic design principles:

- Use HTTP method(s) explicitly (specifically the GET method);
- Be stateless;
- Expose directory structure-like URIs;
- Transfer XML and JSON.

3.1.1. RESTful HTTP mapping

RESTful services typically map the four main HTTP methods to the operations they perform: create (POST), retrieve (GET), update (PUT), and delete (also called DELETE).

Please note, NO's responses to the Federated register will currently use only the HTTP GET method.

3.2. REQUIREMENTS

The RESTful service provided must meet specific business and technical requirements:

- ***Business perspective:***
    - The RESTful service should generate XML, based on a request via an application or publication number;
    - The RESTful service should validate this XML against the XSD provided by EPO,
    - The RESTful service should transform this validated XML into JSONP format before returning it to the client
    - EPO application (or publication) numbers will be used on the client side to request NO data;
    - The NOs are responsible for the business content of the data provided.

- ***Technical perspective:***

  - The service should provide valid JSONP;
  - The default function name for the JSONP output should be "parseResponse";
  - The service should have a high availability (99,5% uptime, 24/7);
  - The service should be fast (< 5 seconds per request);
  - The service should be monitored by the NOs to detect any technical problems. In cases when the service is not available or there are problems detected, the EPO should be informed;
  - Technical support should be available and response time should be within 24 hours of a reported problem.

3.3. COMMON RESPONSE STRUCTURES (XML)

3.3.1. Overview of the schema (XSD)

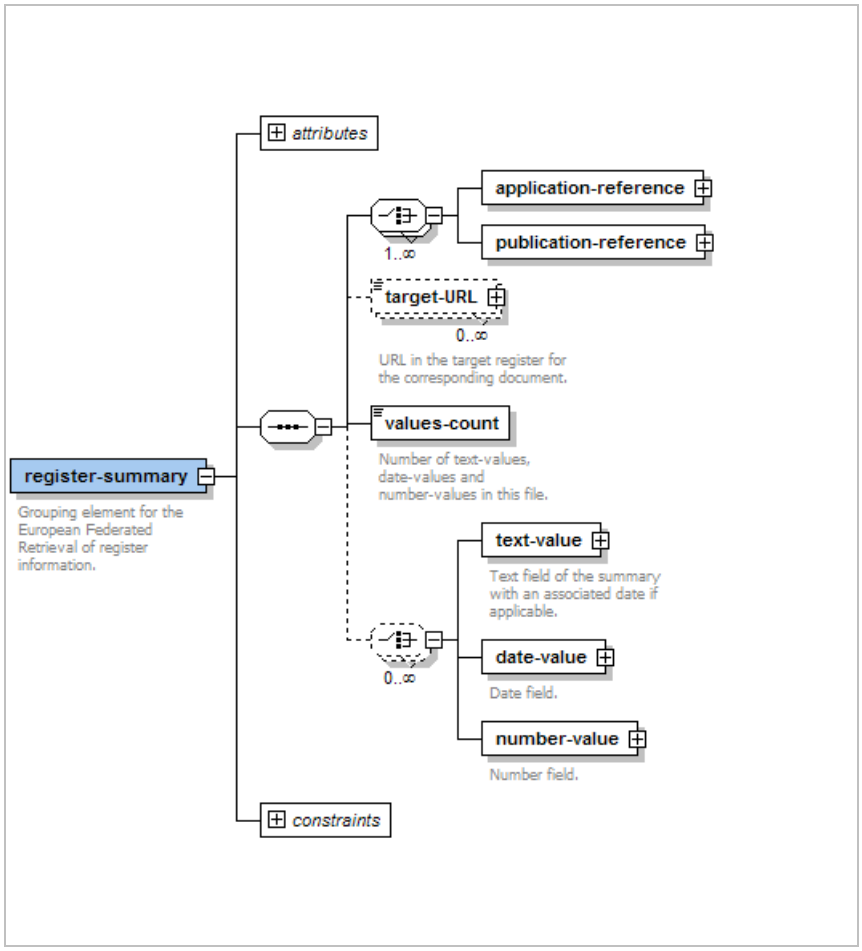


Figure: *XSD overview*

3.3.1.1. Attributes response

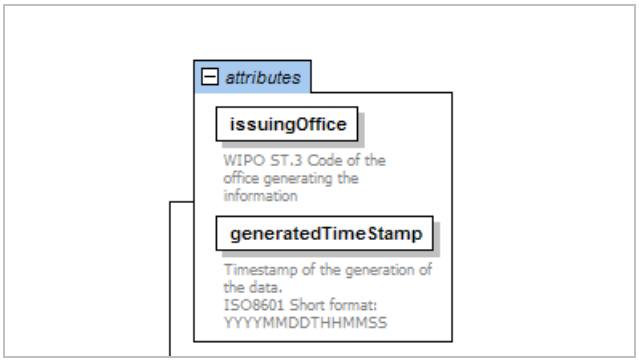


Figure: *attributes*

Attribute Name	Description
`issuingOffice`	Identifies the provider with a Country Code
`generateTimestamp`	Date and time when the XML has been produced

Table: *attributes and descriptions*

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<register-summary issuingOffice="GB"
generatedTimeStamp="20130123T131800">
```

3.3.1.2. Application- / publication-reference response

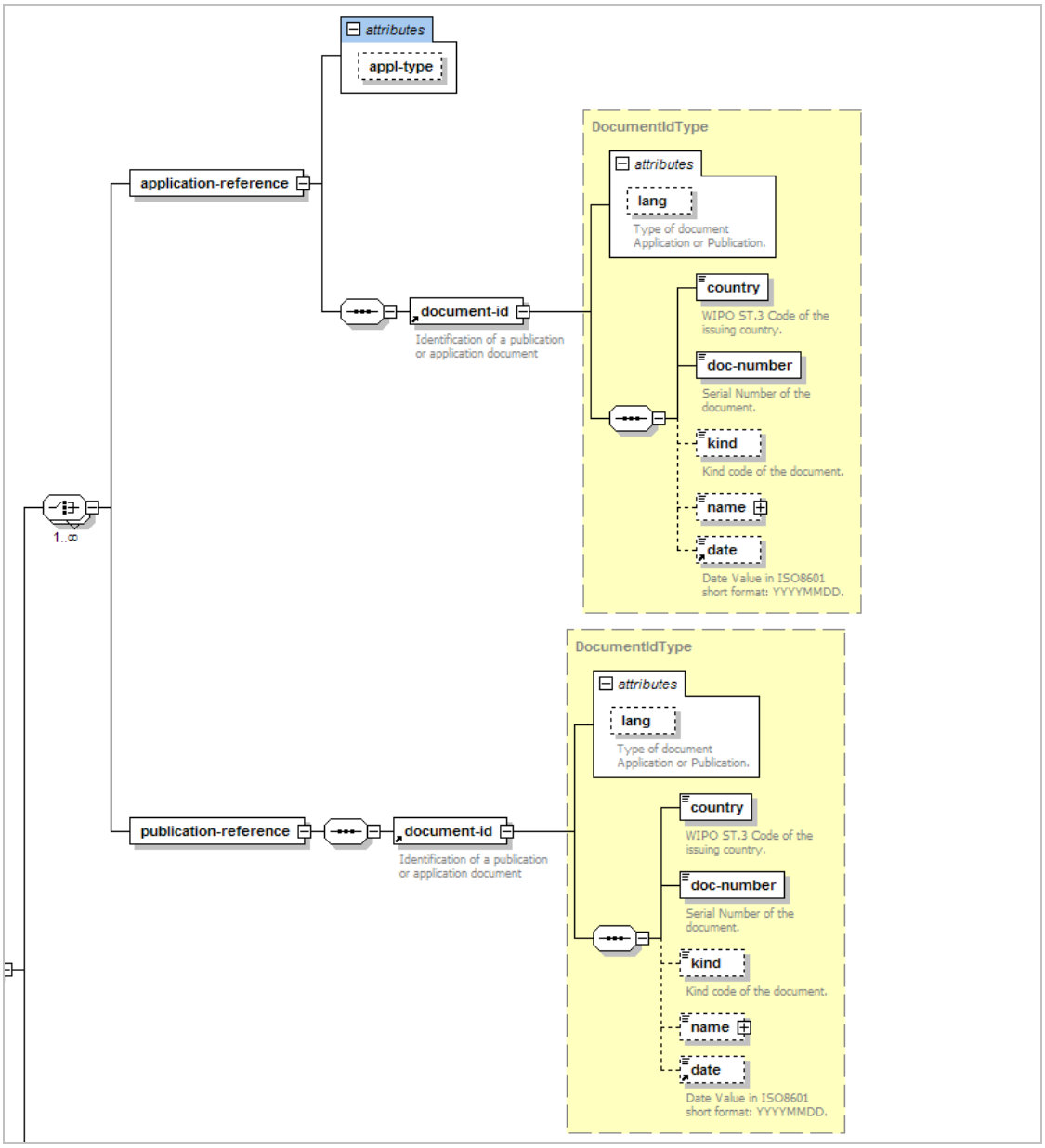


Figure: *application- and publication-reference*

Element	Description
`country`	Country code of the document
`doc-number`	Document number
`kind`	The document's kind code

name	if present, can be "legal" or "natural", i.e. legal should be used for a legal entity whereas natural of a personal (the latter being implied).
date	relevant date for the document: - priority document: filing date - application: filing date - published document: publication date

Table: *application- and publication-reference element descriptions*

3.3.2. Document-id

As you can see in the figure above, all of the reference types use the **document-id** element, which contains the information about the reference:

- country,
- doc-number,
- kind,
- name,
- date

The **document-id** element provides application and publication numbers used by the NOs to identify the validation of an EP patent in the various NO patent registers.

Please note, in the schema above (*§3.3.1.2. Application- / publication-reference response*)**,** the doc-number is **mandatory**.

Example:

```
<application-reference>
        <document-id>
                <country>DE</country>
                <doc-number>69905327.7</doc-number>
                <date>19991108</date>
        </document-id>
</application-reference>
<publication-reference>
        <document-id>
                <country>EP</country>
                <doc-number>1000000</doc-number>
        </document-id>
</publication-reference>
```

3.3.2.1. Target URL

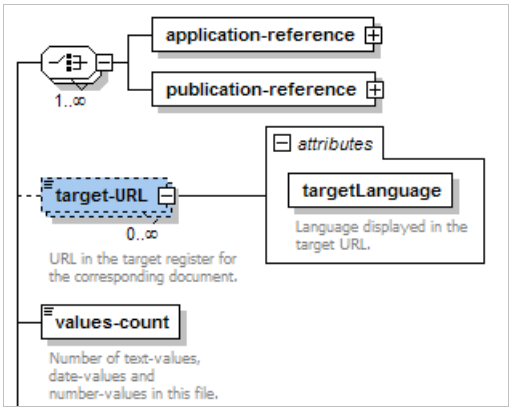


Figure: *target URL*

This information corresponds to the URL that gives direct access to the NO's register page pertaining to a particular patent or application. This URL should be similar to the URL created by the *Deep Linking* project - details on EPOXY (https://epoxy.epo.org).

For **example**, for the **status of patent EP2051816** (EP 99952285.7) in the national phase, the **Netherlands patent office** would return this information.

XML example:

```
<target-URL targetLanguage="en">
http://register.octrooicentrum.nl/register/data/EPNL/99952285E
</target-URL>

<target-URL targetLanguage="nl">
http://register.octrooicentrum.nl/register/gegevens/EPNL/99952285E
</target-URL>
```

JSON example:

```
target-URL => Array (2)
    (
        ['0'] (
            $ = "
```

```
http://register.octrooicentrum.nl/register/data/EPNL/99952285E "
            @targetLanguage = "en"
        )
        ['1'] (
            $ = "
http://register.octrooicentrum.nl/register/gegevens/EPNL/99952285E "
            @targetLanguage = "nl"
        )
    )
```

3.3.2.2. Values-count response

This part of the schema, i.e. the <values-count> tag, is used to check the total number of <text-value>, <date-value>, <code-value> and <number-value> in the file:

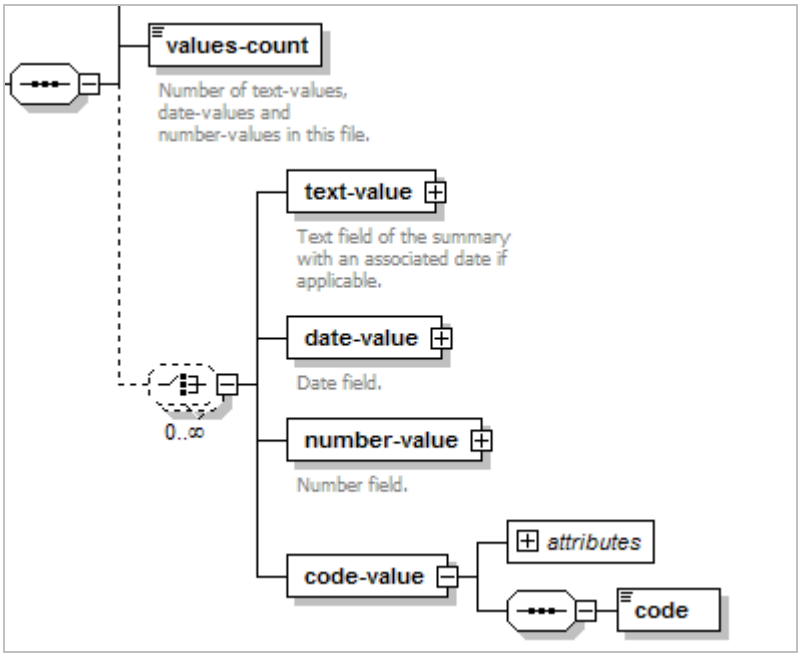


Figure: *values-count response*

The <values-count> tag indicates the total amount of <date-value>, <text-value>, <code-value> and <number-value> tags that are in the retrieved XML. The <values-count> tag will be used to validate the retrieved content.

Example:

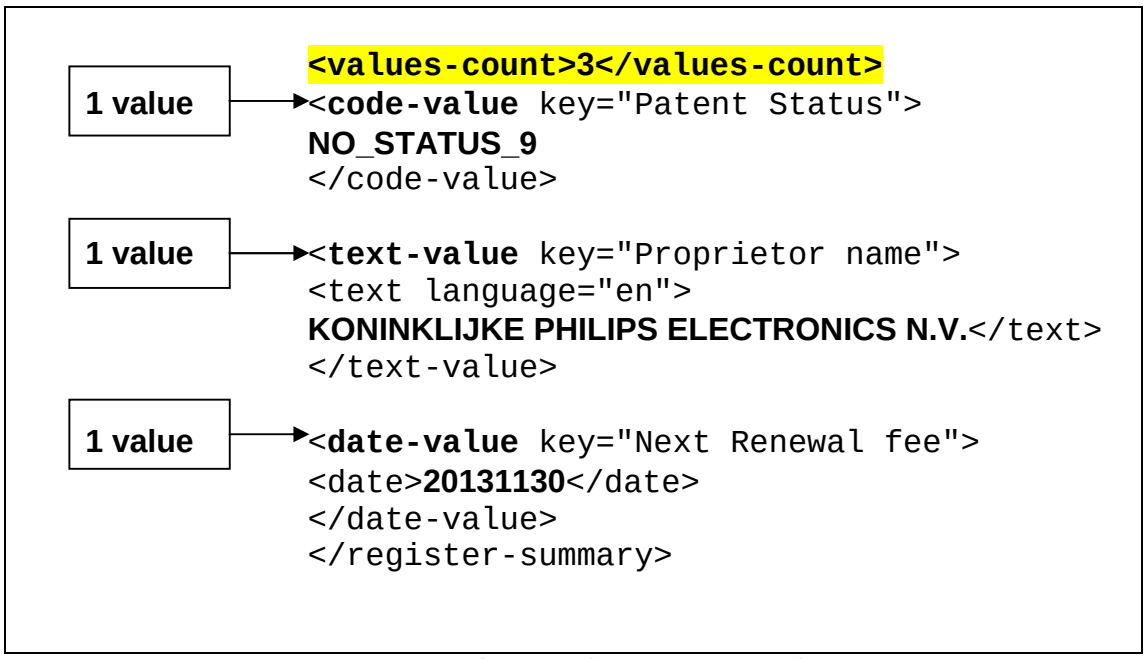


Figure: *values count example*

In the example above you can see that **three** elements were received per the `<values-count>`: one `<code-value>`, one `<text-value>` and one `<date-value>` tag.

3.3.2.3. Text/date/number/code-values response

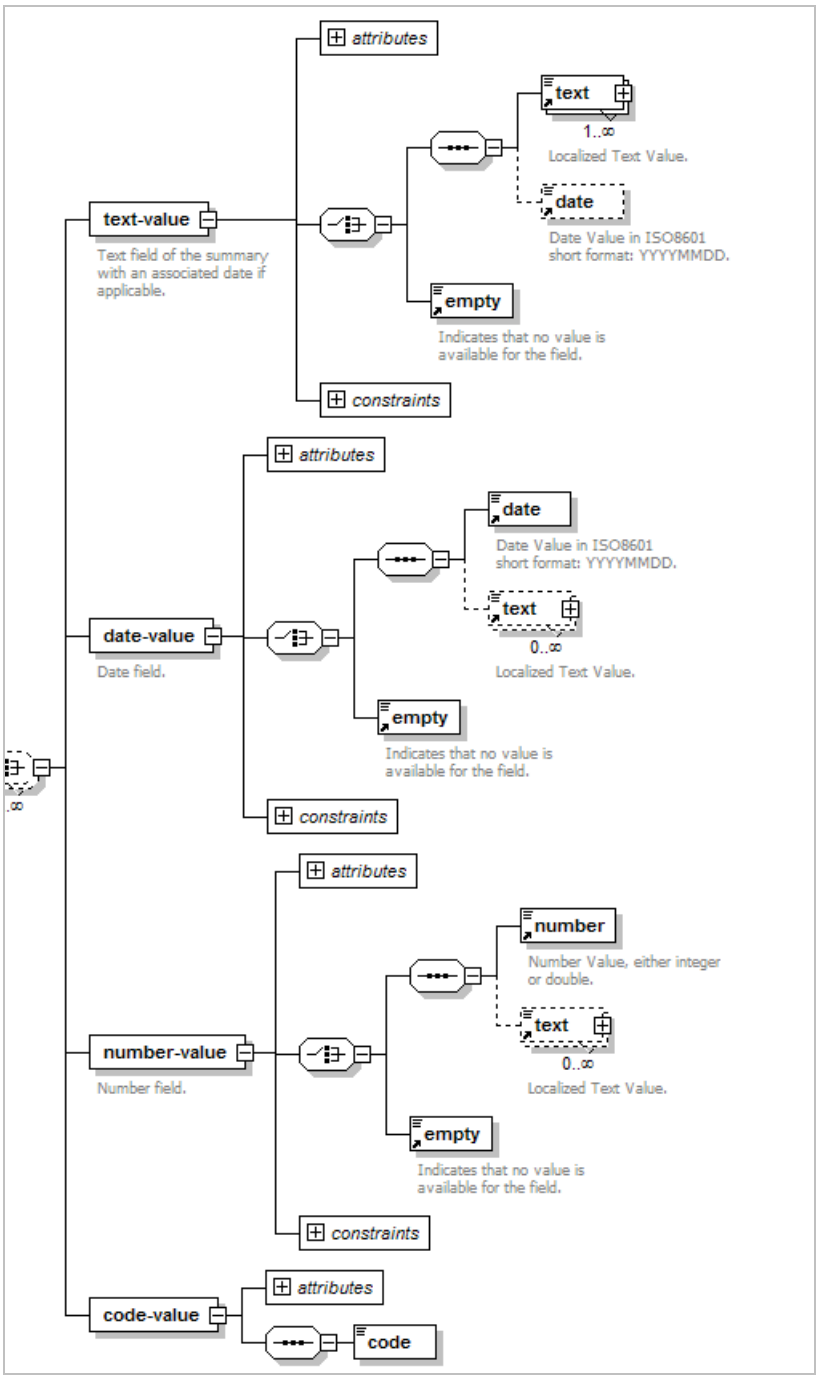


Figure: *text-, date-, code-, number-values response*

Element	Description
`text-value`	Localized text value. The language code is provided within the language attribute of the text element. It can additionally have a date element (e.g. `date-value` below).
`date-value`	Date value in ISO8601 short format: YYYYMMDD. It can additionally have a text element (e.g. `text-value` element above).
`number-value`	Number value, either integer or double. It can additionally have a text element (e.g. `text-value` element above).
`code-value`	Code value corresponds to the key of the Status stored in the CMS. This is the NO_STATUS_X key (**see** the **table "*Status list* *as of 01-07-2013*"** in **§ 2.12**.)

Table: *text-, date-, number-value element descriptions*

Every `<text-value>` contains an attribute language and can have more than one language.

Example:

```
 <text-value key="Renewal Fee Last Paid">
      <text language="fr">Payé pour des années de
protection: 2</text>
      <text language="en">Accepted renewal fee (years): 2
      </text>
      <date>20120418</date>
 </text-value>
```

Text-value, **code-value**, **date-value** and **number-value** should contain a first input (text / date / number). As an option, it can contain a second input (date / text/ text).

Please note, the first input and the second input are always displayed in the Federated Register overview table when received by the client.

Example:

```
<date-value key="Renewal Fee Last Paid">
      <date>20130902</date>
```

```
      <text language="fr">N° de l'annuité payée 14:
380.00€</text>
      <text language="en">N° of the renewal fee 14:
380.00€</text>
</date-value>
```

3.3.3. Data validation

The XML produced by the RESTful service has to be compliant with the schema. In addition, the EPO expects each NO to control the data quality to ensure accuracy of the information displayed in the Federated register.

The JSON produced must be compliant with the BadgerFish convention for translating an XML document into a JSON object (see *§4. Useful links*, for more information about the BadgerFish convention.)

Here you can see an **example of valid XML** that the NO RESTful service is expected to generate:

```
<register-summary issuingOffice="DE"
generatedTimeStamp="20130320T112754">
<publication-reference>
  <document-id>
        <country>EP</country>
        <doc-number>2227493</doc-number>
  </document-id>
</publication-reference>

<application-reference>
  <document-id>
        <country>EP</country>
        <doc-number>602008009889</doc-number>
        </document-id>
</application-reference>

<target-URL
targetLanguage="de">http://register.dpma.de/DP
MAregister</target-URL>

<values-count>5</values-count>

<text-value key="Proprietor Name">
  <text language="de">Proprietor_name</text>
</text-value>

<code-value key="Patent Status">
```

```
            <code>NO_STATUS_9</code>
          </code-value>

          <date-value key="Patent Withdrawn">
            <date>20120219</date>
          </date-value>

          <date-value key="Renewal Fee Last Paid">
            <date>20131130</date>
            <text language="de">Jahresgebuhr fur das 6.
          Jahr</text>
            <text language="en">Annual fee for the 6th
          year</text>
          </date-value>

          <date-value key="Last Update of National
          Register">
            <date>20130123</date>
          </date-value>
</register-summary>
```

Figure: *valid XML*

Here is a **JSON example** that was translated from the above XML according to the BadgerFish convention:

```
          {"register-summary":
          {"@issuingOffice":"DE","@generatedTimeStamp"
          :"20130320T115604",

          "publication-reference":{"document-
          id":{"country":{"$":"EP"},"doc-
          number":{"$":"2227493"}}},

          "application-reference":{"document-
          id":{"country":{"$":"EP"},"doc-
          number":{"$":"602008009889"}}},

          "target-
          URL":{"@targetLanguage":"de","$":"http:\/\/r
          egister.dpma.de\/DPMAregister"},

          "values-count":{"$":"5"},

          "text-value":{"@key":"Proprietor
          Name","text":{"@language":"de","$":"Propriet
          or_name"}},
```

```
"code-
value":{"code":{"$":"NO_STATUS_9"},"@key"
:"Patent Status"},

"date-value":[{"@key":"Patent
Withdrawn","date":{"$":"20120219"}},
{"@key":"Renewal Fee Last
Paid","date":{"$":"20131130"},
"text":[{"@language":"de","$":"Jahresgebuhr
fur das 6. Jahr"},
{"@language":"en","$":"Annual fee for the
6th year"}]},
{"@key":"Last Update of National
Register","date":{"$":"20130123"}}]

}
}
```


Figure: *JSON translated from XML*

3.3.4. JSONP technology

JSONP communication technology will be used to request data from the NO-side. The technology provides a method to request data from a server in a different domain (this is prohibited by web browsers because of the security policy).

The Content-Type in the response header of the RESTful service should be:

`application/javascript;charset=UTF-8`

As you may have concluded, we are kindly requesting that NO's deliver their output in UTF-8 format. This ensures that characters are displayed correctly.

Furthermore, the EPO is expecting NO's to provide a URI with the following parameters:

- `.js` to indicate that it is a JSONP request;
- `callback parameter` to indicate the name of the callback function.

Example:

`https://...../pn/EP2227493.js?callback=`**`parseResponse`**

JSONP example:

```
parseResponse({"register-summary":
{"@issuingOffice":"DE","@generatedTimeStamp":"201
30320T115604",

"publication-reference":{"document-
id":{"country":{"$":"EP"},"doc-
number":{"$":"2227493"}}},

"application-reference":{"document-
id":{"country":{"$":"EP"},"doc-
number":{"$":"602008009889"}}},

"target-
URL":{"@targetLanguage":"de","$":"http:\/\/regist
er.dpma.de\/DPMAregister"},

"values-count":{"$":"5"},

"text-value":[ {"@key":"Proprietor
Name","text":{"@language":"de","$":"Proprietor_na
me"}},
{"@key":"Patent Status","code":{"$":"NO_STATUS_9
"}}],

"date-value":[{"@key":"Patent
Withdrawn","date":{"$":"20120219"}},
{"@key":"Renewal Fee Last
Paid","date":{"$":"20131130"},
"text":[{"@language":"de","$":"Jahresgebuhr fur
das 6. Jahr"},
{"@language":"en","$":"Annual fee for the 6th
year"}]},
{"@key":"Last Update of National
Register","date":{"$":"20130123"}}]
}
}
)
```

Figure: *JSONP example*

In the example above the **`parseResponse(...)`**function is the name of the callback function.

When the query parameter **`callback`** is not provided, the name of the callback function should be **`parseResponse(...)`**

More info about the JSONP technology can be found here:
http://en.wikipedia.org/wiki/JSONP

3.3.5. Correspondence between data display and XML values

The tag values retrieved by a Federated register via the XML file will be aggregated to be displayed according to the table in *§2.5.2. Data from the NOs*.

Here are you can see which tag is used for each item:

XML tags	Data item	Column Headers
`<doc-number> from <application-reference><document-id>`	`application number (number)`	Application No.
`<doc-number> from <publication-reference><document-id>`	`publication number (number)`	Publication No.
`<text-value key="Proprietor Name">`	`proprietor name (text)`	Proprietor
`<code-value key="`**`Patent Status`**`">`	`status (code)`	**Status** (also refer to § 2.12. *Status*)
`<date-value key="`**`Patent Withdrawn`**`">`	`patent withdrawn (date)`	**Invalidation date** (please see table in *§2.5.2. Data from the NOs*.)
`<date-value key="`**`Patent Revoked`**`">`	`patent revoked (date)`	
`<date-value key="`**`Cancelled`**`">`	`patent lapsed (date)`	
`<date-value key="`**`Patent Expired`**`">`	`patent expired (date)`	
`<date-value key="Patent Not In Force">` `optional : <text language="lg">`	`not in force (date)`	Not in force

`<date-value key="Renewal Fee Last Paid">` `optional : <text language="lg">`	`renewal fees last paid (date)`	Renewal fees last paid
`<date-value key="Last Update of National Register">`	`last update of the nat. register (date)`	Register last updated

Table: *tags, data items, column headers*

3.3.6. RESTful service example (built by the EPO)

As stated in §1. *Introduction*, each NO decides individually which clients and technologies to use when building their RESTful service.

However, here is an example to use as a guideline:

Description	Element and attribute	Type	Format

Application number	`<application-reference>`
	`<document-id>`
	`<country>DE</country>`
	`<doc-number>`**`{VALUE}`**`</doc-number>`
	`</document-id>`
	`</application-reference>`
		text	yynnnnnn **(8)**
			
			Note that you have to specify the country as well, in order to match your full application number, e.g. to get DE69934528, you will have to specify `<country>DE</country>`

Publication number	`<publication-reference>`
	`<document-id>`
	`<country>EP</country>`
	`<doc-number>`**`{VALUE}`**`</doc-number>`
	`</document-id>`
	`</publication-reference>`
		text	nnnnnnn (**7**)

Proprietor name	`<text-value key="Proprietor Name">`
	`<text language="{LANGUAGE_CODE}">`**`{VALUE}`**`</text>`
	`</text-value>`
		text	**Maximum 150 characters**

Status	`<code-value key="Patent Status">`
	`<code>`**`{VALUE}`**`</code>`
	`</code-value>`
		text	Please refer to the table in § 2.12. *Status*.

Patent withdrawn	`<date-value key="Patent Withdrawn">`	date	All dates must use the ISO8601 short format

(order: **1st** [1])	<date>**{VALUE}**</date> </date-value>		Time Stamp: **YYYYMMDD**
Patent revoked (order: **2nd** [1])	<date-value key="Patent Revoked"> <date>**{VALUE}**</date> </date-value>	date	
Patent lapsed (order: **3rd** [1])	<date-value key="Patent Cancelled"> <date>**{VALUE}**</date> </date-value>	date	
Patent expired (order: **4th** [1])	<date-value key="Patent Expired"> <date>**{VALUE}**</date> </date-value>	date	
Not in force	<date-value key="Patent Not In Force"> <date>**{VALUE}**</date> </date-value>	date	
Renewal fees last paid	<date-value key="Renewal Fee Last Paid"> <date>**{VALUE}**</date> </date-value>	date	
Last update of the national register	<date-value key="Last Update of National Register"> <date>**{VALUE}**</date> </date-value>	date	
URL of NO register	<target-URL targetLanguage="en">**{VALUE}**</target-URL>	text	
Number of text-values, date-values and number-values in this file	<values-count>**{VALUE}**</values-count>	integer	This number should match the amount of text/date/number values that are below this element.
Name of issuing office	element: register-summary attribute: issuingOffice="{VALUE}"	text	Value should be in capitals and use one of the WIPO ST.3 country codes (for reference see the WIPO ST.3 Annex A).
Time stamp of generation of page	element: register-summary attribute: generatedTimeStamp="{VALUE}"	date	Value should be in ISO8601 Short format: **YYYYMMDDTHHMMSS**

			example: 20130123T124800

Table: *RESTful examples*

[1] If more than 1 item is returned, the value with the highest order will be displayed.

An XML example file, containing the maximum amount of elements, can be found here.

Please note, if there is no data for a field, the field must be omitted.

4. USEFUL LINKS

BadgerFish
http://badgerfish.ning.com/

Epoxy
https://epoxy.epo.org/

HTTP safe methods
http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Safe_methods

JSONP technology
http://en.wikipedia.org/wiki/JSONP

REST: What is it?
http://searchsoa.techtarget.com/definition/REST

RESTful Web services: The basics
http://www.ibm.com/developerworks/webservices/library/ws-restful/

W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures
http://www.w3.org/TR/xmlschema11-1/

XML Schema Part 2: Datatypes Second Edition
http://www.w3.org/TR/xmlschema-2/