



COMPONENT

Project Acronym: OpenUp!
Grant Agreement No: 270890
Project Title: Opening up the Natural History Heritage for Europeana

C5.1.1: Data quality service for botanical names prototype

Revision: 1.3

Authors:

Nick Black, Royal Botanic Gardens, Kew (RBGK -16)

| Project co-funded by the European Commission within the ICT Policy Support Programme | | |
|--|--|---|
| Dissemination Level | | |
| P | Public | x |
| C | Confidential, only for members of the consortium and the Commission Services | |



REVISION HISTORY AND STATEMENT OF ORIGINALITY

Revision History

| Revision | Date | Author | Organisation | Description |
|-----------------|-------------|---------------|---------------------|---|
| 1.1 | 22/08/2011 | N. Black | RBGK | |
| 1.2 | 23/08/2011 | S. Hargreaves | RBGK | Formatting |
| 1.3 | 26/08/2011 | A.Saltmarsh | RBGK | Reviewed by Technology Management Group |
| | 05/09/2011 | PB | BGBM | editing |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Statement of Originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Distribution

| Recipient | Date | Version | Accepted YES/NO |
|---------------------|--------------|----------------|------------------------|
| Karol Marhold | 02/09/2011 | 1.3 | yes |
| Project Coordinator | 05 Sept 2011 | 1.3c | yes |



Table of Contents

| | |
|----------------------------------|---|
| 1. Introduction | 1 |
| 2. Enhanced Taxonomic check..... | 1 |
| a. Introduction | 1 |
| b. Metadata..... | 1 |
| c. Single query mode | 2 |
| d. Multiple query mode | 4 |
| e. Specifying extra data..... | 6 |
| List of References..... | 6 |



1. Introduction

Following on from the initial release of the botanical data quality service, the nomenclatural check has been updated to return the taxonomic status of the name along with its accepted name or synonyms. As before the name and concept check are each implemented as a HTTP accessed JSON format webservice, and are available from the following URL: <http://data.kew.org/openup-data-quality-service/> Please note the new version of the service will not be available until 31st August 2011.

2. Enhanced Taxonomic check

a. Introduction

For the prototype to demonstrate a taxonomic check the service now queries 'The Plant List' database - <http://www.theplantlist.org/> . As in the initial release the service accepts a name (or names) and returns taxonomic status along with accepted name or synonyms.

The **status** flag will be one of A (accepted), S (synonym), U (unplaced).

Note: For this version of the prototype the query string must include a valid Genus and Species. Fuzzy matching and name parsing have not been implemented.

b. Metadata

The service must support a **metadata request** - called by requesting the reconciliation service URL with no parameters or just a callback function name parameter.

Request

Sample URL: <http://data.kew.org/openup-data-quality-service/concept-reconciler>

Response

```
{
  "name" : "Kew concept reconciliation service",
  "identifierSpace" : "http://data.kew.org",
  "schemaSpace" : "http://data.kew.org" ,
}
```

Request (using callbacks)

Sample URL: http://data.kew.org/openup-data-quality-service/concept-reconciler?callback=my_function



Response (using callbacks)

```
my_function(  
  
  {  
    "name" : "Kew concept reconciliation service",  
    "identifierSpace" : "http://data.kew.org",  
    "schemaSpace" : "http://data.kew.org" ,  
  }  
  
)
```

c. Single query mode

Request

Queries are specified in JSON. A single query can be specified as follows:

```
{  
  "query" : "Ocimum americanum",  
  "limit" : 3,  
  "type" : "concept",  
}
```

The JSON query is specified as a URL parameter, and the service can respond to either GET or POST.

Response

```
{  
  result : [  
  
    {  
      id: "kew-136802"  
      , name: "Ocimum americanum L."  
      , type: ["concept"]  
      , score: "100"  
      , match: "true"  
      , status: "A"  
  
      , synonyms: [  
  
        "[Ocimum album Roxb."  
        ,  
        " Ocimum brachiatum Blume"  
        ,  
        " Ocimum canum Sims"  
        ,  
        " Ocimum dichotomum Hochst. ex Benth."  
        ,  
      ]  
    }  
  ]  
}
```



```
        " Ocimum dinteri Briq."
        ,
        " Ocimum fluminense Vell."
        ,
        " Ocimum fruticulosum Burch."
        ,
        " Ocimum hispidulum Schumach. & Thonn."
        ,
        " Ocimum incanescens Mart."
        ,
        " Ocimum stamineum Sims"
        ,
        " Ocimum thymoides Baker]"
    ]

    }
    ,
    {
    id: "kew-136804"
    , name: "Ocimum americanum var. pilosum (Willd.) A.J.Paton"
    , type: ["concept"]
    , score: "100"
    , match: "true"
    , status: "S"

    , accepted_names: [
        "Ocimum africanum"
    ]
    }
    ]
}
```

As per the metadata request, a callback function name can be specified - this will have the effect of wrapping the JSON response in a function call to the specified function name.



d. Multiple query mode

Here, multiple queries are passed in a single request. The individual queries are specified as above, but are keyed. These keys are used to link the results to the queries.

Request

```
{ "q1" : {
  "query" : "Ocimum americanum",
  "limit" : 3,
  "type" : "concept",
}
, "q2" : {
  "query" : "Ocimum africanum",
  "limit" : 3,
  "type" : "concept",
} } }
```

Response

```
{
  q2 : {
    result : [
      {
        id: "kew-136798"
        , name: "Ocimum × africanum Lour."
        , type: ["concept"]
        , score: "100"
        , match: "true"
        , status: "A"
        , synonyms: [
          "[Ocimum americanum var. pilosum (Willd.) A.J.Paton"
          ,
          " Ocimum basilicum var. anisatum Benth."
          ,
          " Ocimum basilicum var. pilosum (Willd.) Benth."
          ,
          " Ocimum × citriodorum Vis."
          ,
          " Ocimum × graveolens A.Br."
          ,
          " Ocimum × petitianum A.Rich."
          ,
          " Ocimum × pilosum Willd.]"
        ]
      }
    ]
  }
}
```



```
    ]
  }
,
q1 : {
  result : [
    {
      id: "kew-136802"
      , name: "Ocimum americanum L."
      , type: ["concept"]
      , score: "100"
      , match: "true"
      , status: "A"
      , synonyms: [
          "[Ocimum album Roxb."
          ,
          " Ocimum brachiatum Blume"
          ,
          " Ocimum canum Sims"
          ,
          " Ocimum dichotomum Hochst. ex Benth."
          ,
          " Ocimum dinteri Briq."
          ,
          " Ocimum fluminense Vell."
          ,
          " Ocimum fruticosum Burch."
          ,
          " Ocimum hispidulum Schumach. & Thonn."
          ,
          " Ocimum incanescens Mart."
          ,
          " Ocimum stamineum Sims"
          ,
          " Ocimum thymoides Baker]"
        ]
    }
  ]
}
```




```
    }
  ,
  {
    id: "kew-136804"
    , name: "Ocimum americanum var. pilosum (Willd.) A.J.Paton"
    , type: ["concept"]
    , score: "100"
    , match: "true"
    , status: "S"

    , accepted_names: [

      "Ocimum africanum"

    ]
  }
]
}}
```

As per the metadata and single query requests, a callback function name can be specified.

e. Specifying extra data

The query object can also include a properties array, which is a way to send extra data to the service. This could be used to optionally include fields such as family name and authorship, or to supply a name id to which the name has already been matched.

```
{
  "query": "Ocimum gratissimum var. gratissimum",
  "type": "concept",
  "type_strict": "should",
  "properties": [ {
    "pid": "family",
    "v": "Lamiaceae"
  }, {
    "pid": "nameid",
    "v": "urn:lsid:ipni.org:names:12345-1"
  }
]
```

Note: Specifying extra data has not been implemented for this version of the prototype.

List of References

1. Google Refine: <http://code.google.com/p/google-refine/>
2. Wikipedia entry for JSONP: <http://en.wikipedia.org/wiki/JSONP>
3. The Plant list: <http://www.theplantlist.org/>