

Turning Passive APIs into Active APIs

Oliver Wolf



Oliver Wolf
@owolf



www.innoQ.com
@innoQ

The German Government's shiny new Open Data Portal

Seite - GovData

https://www.govdata.de

innoQ Add Google Bookmark

Dokumente Apps Neues Bibliothek Das Portal Anmelden

Herzlich willkommen

Es ist soweit! GovData, das Datenportal für Deutschland, ist nun online gegangen. Bis 2014 soll das Portal testweise betrieben und dabei schrittweise ausgebaut und optimiert werden. Weitere Informationen zum Portal finden Sie unter "[Das Portal](#)".

Suchen Sie hier nach Daten, Dokumenten und Apps.

Stöbern Sie in diesen Kategorien durch Daten, Dokumente und Apps

 Bevölkerung (1641)	 Bildung und Wissenschaft (32)	 Geographie, Geologie und Geobasisdaten (113)	 Gesetze und Justiz (7)	 Gesundheit (68)	 Infrastruktur, Bauen und Wohnen (1629)	 Kultur, Freizeit, Sport und Tourismus (77)
 Öffentliche Verwaltung, Haushalt und Steuern (162)	 Politik und Wahlen (36)	 Soziales (58)	 Transport und Verkehr (257)	 Umwelt und Klima (137)	 Verbraucherschutz (6)	 Wirtschaft und Arbeit (566)

Neueste Datensätze

Neueste Dokumente

Neueste Apps

bonn

Filtern nach...

4 Treffer

Typ

☐ Datensatz (4)

Offenheit der Lizenz

☒ Alle☐ Freie Nutzung (4)

Kategorien

☐ Öffentliche Verwaltung, Haushalt und Steuern (2)☐ Umwelt und Klima (1)☐ Kultur, Freizeit, Sport und Tourismus (1)

Schlüsselworte

☐ bonn (3)☐ Pflanzenart (1)☐ Ornithologie (1)☐ Niedermoor (1)☐ Naturschutzprogramm (1)

Formate

☐ JSON (3)☐ WMS (1)

Lizenzen

☐ Creative Commons
Namensnennung (CC-BY) (3)

Sortieren nach

Relevanz

Name

Änderung

1

 Pressemittelungen Bonn

API, mit der die Pressemittelungen der Bundesstadt Bonn abgefragt werden können

Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Info...

Kategorien: Öffentliche Verwaltung, Haushalt und Steuern

JSON

Freie Nutzung

Letzte Änderung:
28.2.2013 **Veranstaltungskalender Bonn**

API, mit der alle Veranstaltungen der Bundesstadt Bonn abgefragt werden können

Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Info...

Kategorien: Kultur, Freizeit, Sport und Tourismus

JSON

Freie Nutzung

Letzte Änderung:
28.2.2013 **Bürgeranliegen online Bonn**

REST-API, mit der die Anliegen- und Dienstleistungskategorien (Services) und eingebrachten Bürgeranliegen (Requests) der Bundesstadt Bonn abgefragt werden können

Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Info...

Kategorien: Öffentliche Verwaltung, Haushalt und Steuern

JSON

Freie Nutzung

Letzte Änderung:
28.2.2013 **Kartendienst Naturschutz**

Der Kartendienst stellt folgende Informationen zur Verfügung: Bestandserfassung für den Naturschutz im Laufe des Jahres werden vom Niedersächsischen Landesbetrieb für Wasserwirtschaft,...

Veröffentlichende Stelle: Niedersächsisches Ministerium für Umwelt, Energ...

Kategorien: Umwelt und Klima

WMS

Freie Nutzung

Letzte Änderung:
5.3.2013

bonn



Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Informationstechnologie

**Bürgeranliegen online Bonn**

REST-API, mit der die Anliegen- und Dienstleistungskategorien (Services) und eingebrachten Bürgeranliegen (Requests) der Bundesstadt Bonn abgefragt werden können

[Klicken Sie hier für weitere Informationen](#)

Links zu den Datendateien:

[Liste aller Services](#)

JSON

[Liste aller Requests](#)

JSON

Kategorien: [Öffentliche Verwaltung](#), [Haushalt und Steuern](#)Schlüsselwörter: [anliegen](#), [bonn](#), [bürger](#), [meldung](#)**Bewertung abgeben**

Sie müssen angemeldet sein, um Bewertungen abgeben zu können.

Kommentare erstellen und ansehen

Sie müssen angemeldet sein, um Kommentare verfassen zu können.

Freie Nutzung

Nutzungsbedingungen:

[Creative Commons Namensnennung \(CC-BY\)](#)

Stand Metadaten:

28.2.2013

Veröffentlicht:

21.11.2012

Zeitraum:

22.4.2013 - 22.4.2013

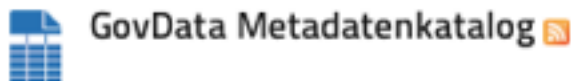
☆☆☆☆ (0)

API?

GovData ► Daten

In dieser Rubrik können Sie nach Datensätzen suchen. Die über GovData verfügbaren Datensätze liegen in maschinenlesbaren, weiterverarbeitbaren Dateiformaten vor.

Veröffentlichende Stelle: Bundesministerium des Inneren



Metadaten werden bei GovData in CKAN verwaltet. Sie können ohne Anmeldung über die Schnittstelle ausgelesen werden. Die Metadaten Struktur ist hier beschrieben: <http://www.govdata.de/metadaten/schema>. Die CKAN API ist hier dokumentiert: <http://docs.ckan.org/en/latest/api.html>. Derzeit wird CKAN 1.8 verwendet.

[Klicken Sie hier für weitere Informationen](#)

Links zu den Datendateien:

[CKAN v1.8 API](#)

JSON

Freie Nutzung

Nutzungsbedingungen:
[Datenlizenz Deutschland](#)
[Namensnennung](#)

Stand Metadaten:
1.3.2013

Veröffentlicht:
19.2.2013

Zeitraum:
22.4.2013 - 22.4.2013

<https://www.govdata.de/ckan/api>

Kategorien: Öffentliche Verwaltung, Wirtschaft und Soziales

Schlüsselwörter: [api](#), [ckan](#), [datensätze](#), [katalog](#), [metadaten](#)

Bewertung abgeben

Sie müssen angemeldet sein, um Bewertungen abgeben zu können.

Kommentare erstellen und ansehen

Sie müssen angemeldet sein, um Kommentare verfassen zu können.

GET /**ckan/api** HTTP/1.1

Host: www.govdata.de

Accept: */*

HTTP/1.1 200 OK

Content-Type: application/json; charset=utf-8

{"version": 1}



The Action API — CKAN Documentation 2.1a documentation

docs.ckan.org/en/latest/api.html

CKAN Documentation v2.1a

The Action API

CKAN's Action API is a powerful, RPC-style API that exposes all of CKAN's core features to API clients. All of a CKAN website's core functionality (everything you can do with the web interface and more) can be used by external applications that call the CKAN API. For example, using the CKAN API your application can:

- Get JSON-formatted lists of a site's datasets, groups or other CKAN objects:
 - http://demo.ckan.org/api/3/action/package_list
 - http://demo.ckan.org/api/3/action/group_list
 - http://demo.ckan.org/api/3/action/tag_list
- Get a full JSON representation of a dataset, resource or other object:
 - http://demo.ckan.org/api/3/action/package_show?id=adur_district_spending
 - http://demo.ckan.org/api/3/action/tag_show?id=gold
 - http://demo.ckan.org/api/3/action/group_show?id=data-explorer
- Search for packages or resources matching a query:
 - http://demo.ckan.org/api/3/action/package_search?q=spending
 - http://demo.ckan.org/api/3/action/resource_search?query=name:District%20Names
- Create, update and delete datasets, resources and other objects
- Get an activity stream of recently changed datasets on a site:
 - http://demo.ckan.org/api/3/action/recently_changed_packages_activity_list

Making an API Request

To call the CKAN API, post a JSON dictionary in an HTTP POST request to one of CKAN's API URLs. The parameters for the API function should be given in the JSON dictionary. CKAN will also return its response in a JSON dictionary.

One way to post a JSON dictionary to a URL is using the command-line HTTP client [HTTPie](#). For example, to get a list of the names of all the datasets in the [data-explorer](#) group on demo.ckan.org, install HTTPie and then call the [group_list](#) API function by running this command in a terminal:

```
http http://demo.ckan.org/api/3/action/group_list id=data-explorer
```

The response from CKAN will look like this:

On this Page

- The Action API
- Making an API Request
- API Versions

Table of Contents

- Option 1: Package Installation
- Comments and Commenting
- Add Extensions
- Linked Data and RDF
- FileStore and File Uploads
- DataStore
- Background Tasks
- Import ("Harvest") Data from Other Sites
- Geospatial Capabilities
- Translating Datasets, Groups and Tags
- Send Email Notifications to Users
- Page View Tracking
- Publishing Datasets
- Load Datasets
- Set and Manage Permissions
- Publisher Profile and Workflow
- Data Viewer
- The Action API**
- Making an API Request

Operations on collections

Operations on individual objects/entities

Search/Query operations

bonn

Filtern nach...

4 Treffer

Typ

☐ Datensatz (4)

Offenheit der Lizenz

☒ Alle☐ Freie Nutzung (4)

Kategorien

- ☐ Öffentliche Verwaltung, Haushalt und Steuern (2)
- ☐ Umwelt und Klima (1)
- ☐ Kultur, Freizeit, Sport und Tourismus (1)

Schlüsselworte

- ☐ bonn (3)
- ☐ Pflanzenart (1)
- ☐ Ornithologie (1)
- ☐ Niedermoor (1)
- ☐ Naturschutzprogramm (1)

Formate

- ☐ JSON (3)
- ☐ WMS (1)

Lizenzen

- ☐ Creative Commons Namensnennung (CC-BY) (3)

Sortieren nach Relevanz Name Änderung

1



Pressemittelungen Bonn

API, mit der die Pressemittelungen der Bundesstadt Bonn abgefragt werden können

Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Info...

Kategorien: Öffentliche Verwaltung, Haushalt und Steuern

JSON

Freie Nutzung

Letzte Änderung:
28.2.2013

Veranstaltungskalender Bonn

API, mit der alle Veranstaltungen der Bundesstadt Bonn abgefragt werden können

Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Info...

Kategorien: Kultur, Freizeit, Sport und Tourismus

JSON

Freie Nutzung

Letzte Änderung:
28.2.2013

Bürgeranliegen online Bonn

REST-API, mit der die Anliegen- und Dienstleistungskategorien (Services) und eingebrachten Bürgeranliegen (Requests) der Bundesstadt Bonn abgefragt werden können

Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Info...

Kategorien: Öffentliche Verwaltung, Haushalt und Steuern

JSON

Freie Nutzung

Letzte Änderung:
5.3.2013

Kartendienst Naturschutz

Der Kartendienst stellt folgende Informationen zur Verfügung: Bestandserfassung für den Naturschutz im Laufe des Jahres werden vom Niedersächsischen Landesbetrieb für Wasserwirtschaft,...

Veröffentlichende Stelle: Niedersächsisches Ministerium für Umwelt, Energ...

Kategorien: Umwelt und Klima

WMS

Freie Nutzung

Package ("Datensatz")

bonn



Veröffentlichende Stelle: Bundesstadt Bonn, Amt für Organisation und Informationstechnologie

Freie Nutzung

**Bürgeranliegen online Bonn**

REST-API, mit der die Anliegen- und Dienstleistungskategorien (Services) und eingebrachte Bürgeranliegen der Bundesstadt Bonn abgefragt werden können

[Klicken Sie hier für weitere Informationen](#)

Links zu den Datendateien:

[Liste aller Services](#)

JSON

[Liste aller Requests](#)

JSON

Stand Metadaten:
28.2.2013

Kategorien: Öffentliche Verwaltung, Haushalt und Steuern

Schlüsselwörter: anliegen, bonn, bürger, meldung

Bewertung abgeben

Sie müssen angemeldet sein, um Bewertungen abgeben zu können.

Kommentare erstellen und ansehen

Sie müssen angemeldet sein, um Kommentare verfassen zu können.

Resources ("Datendateien")**Groups ("Kategorien")****Tags ("Schlüsselwörter")**

```
{ "version": 1 }
```

Characteristics of a Good API

(according to Joshua Bloch)

- ▶ **Easy to learn**
- ▶ **Easy to use, even without documentation**
- ▶ **Hard to misuse**
- ▶ **Easy to read and maintain code that uses it**
- ▶ **Sufficiently powerful to satisfy requirements**
- ▶ **Easy to extend**
- ▶ **Appropriate to the audience**

```
{ "version": 1 }
```

Home Documents for HTTP APIs

(draft-nottingham-json-home-02)

Simple canonical JSON representation for HTTP API entry points (“home documents”)

GET / HTTP/1.1
Host: example.org
Accept: application/json-home

HTTP/1.1 200 OK
Cache-Control: max-age=3600
Content-Type: application/json-home

Cacheable on client side

Special media type

Link relation type

Relative URI

Direct link (1 resource)

```
{  
  "resources": {  
    "http://example.org/rel/widgets": {  
      "href": "/widgets/",  
    },  
    "http://example.org/rel/widget": {  
      "href-template": "/widgets/{widget_id}",  
      "href-vars": {  
        "widget_id": "http://example.org/param/widget"  
      },  
    },  
    "hints": {  
      "allow": ["GET", "PUT", "DELETE", "PATCH"],  
      "representations": ["application/json"],  
      "accept-patch": ["application/json-patch"],  
      "accept-post": ["application/xml"],  
      "accept-ranges": ["bytes"]  
    }  
  }  
}
```

Web Linking

(RFC 5988)

Link Relation Types identify the semantics of a link

Two types of Relation Types:

Registered

Identified by a registered token, currently one of:

alternate, appendix, bookmark, chapter, contents, copyright, current, describedby, edit, edit-media, enclosure, first, glossary, help, hub, index, last, latest-version, license, next, next-archive, payment, prev, predecessor-version, previous, prev-archive, related, replies, section, self, service, start, stylesheet, subsection, successor-version, up, version-history, via, working-copy, working-copy-of

Extension

Uniquely identified by a URI which can (but doesn't have to) point to a document describing the link semantics

Simple canonical JSON representation for HTTP API entry points (“home documents”)

```
GET / HTTP/1.1
Host: example.org
Accept: application/json-home
```

```
HTTP/1.1 200 OK
Cache-Control: max-age=3600
Content-Type: application/json-home
```

Cacheable on client side

Special media type

Link relation type

Relative URI

Direct link (1 resource)

Link relation type

Link template

Template variables

Templated link
(0..n resources)

```
{
  "resources": {
    "http://example.org/rel/widgets": {
      "href": "/widgets/",
    },
    "http://example.org/rel/widget": {
      "href-template": "/widgets/{widget_id}",
      "href-vars": {
        "widget_id": "http://example.org/param/widget"
      },
      "hints": {
        "allow": ["GET", "PUT", "DELETE", "PATCH"],
        "representations": ["application/json"],
        "accept-patch": ["application/json-patch"],
        "accept-post": ["application/xml"],
        "accept-ranges": ["bytes"]
      }
    }
  }
}
```

URI Template

(RFC 6570)

URI Templates describe a range of URIs through variable expansion

```
http://example.com/~fred/  
http://example.com/~mark/
```

Examples

Template

```
http://example.com/~{username}/
```

```
http://example.com/dictionary/c/cat  
http://example.com/dictionary/d/dog
```

Examples

Template

```
http://example.com/dictionary/{term:1}/{term}
```

```
http://example.com/search?q=cat&lang=en  
http://example.com/search?q=chien&lang=fr
```

Examples

Template

```
http://example.com/search{?q,lang}
```

(The expression syntax is a little bit richer than this, actually.)

Simple canonical JSON representation for HTTP API entry points (“home documents”)

```
GET / HTTP/1.1
Host: example.org
Accept: application/json-home

HTTP/1.1 200 OK
Cache-Control: max-age=3600
Content-Type: application/json-home
```

Cacheable on client side

Special media type

Link relation type

Relative URI

Direct link (1 resource)

Link relation type

Link template

Template variables

Templated link
(0..n resources)

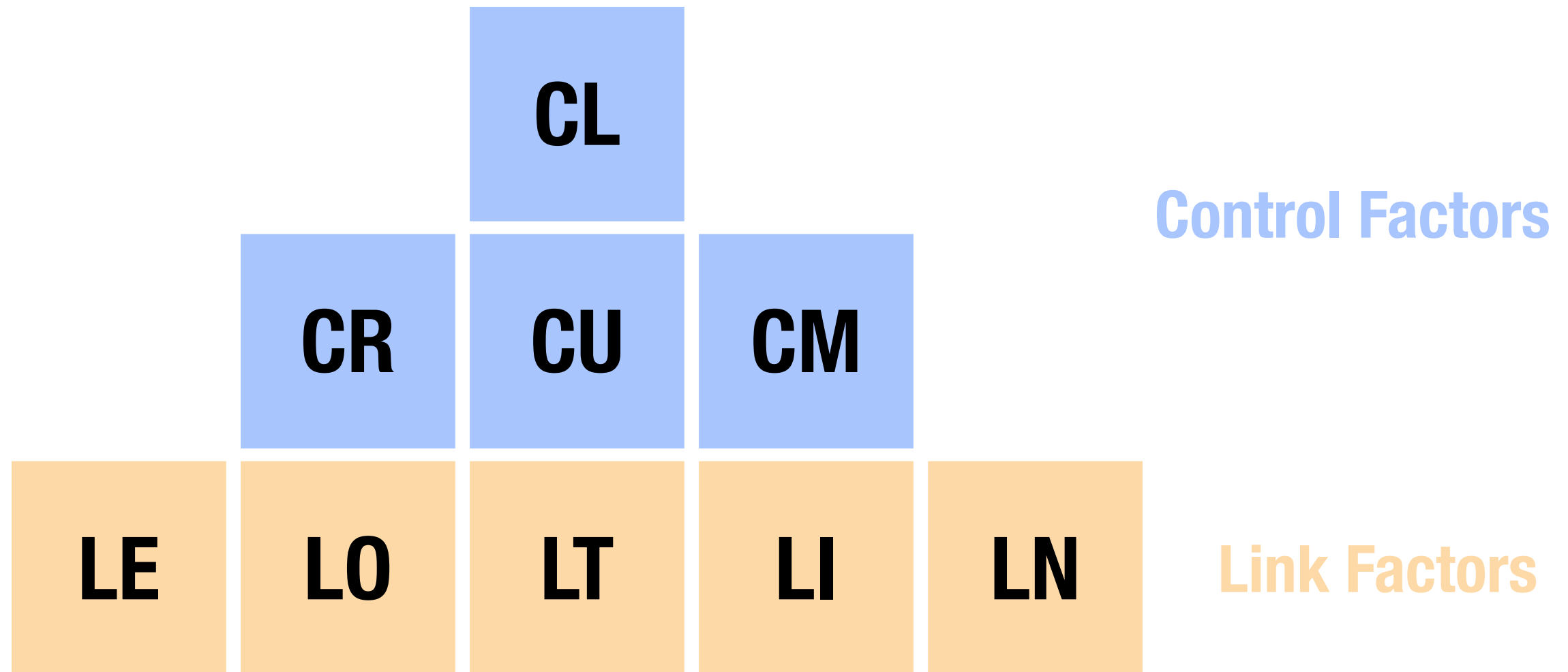
Hints

```
{
  "resources": {
    "http://example.org/rel/widgets": {
      "href": "/widgets/",
    },
    "http://example.org/rel/widget": {
      "href-template": "/widgets/{widget_id}",
      "href-vars": {
        "widget_id": "http://example.org/param/widget"
      },
      "hints": {
        "allow": ["GET", "PUT", "DELETE", "PATCH"],
        "representations": ["application/json"],
        "accept-patch": ["application/json-patch"],
        "accept-post": ["application/xml"],
        "accept-ranges": ["bytes"]
      }
    }
  }
}
```

```
{ "version": 1 }
```

```
{
  "resources": {
    "http://govdata.de/rel/packages": {
      "href": "/action/package_list"
    },
    "http://govdata.de/rel/package": {
      "href-template": "/action/package_show{?id}",
      "href-vars": {
        "id": "http://govdata.de/param/package"
      },
    },
    "http://govdata.de/rel/package_search": {
      "href-template": "/action/package_search{?q,sort,rows,start}",
      "href-vars": {
        "q": "http://govdata.de/param/solr_query",
        "sort": "http://govdata.de/param/solr_sort_criteria",
        "rows": "http://govdata.de/param/max_results",
        "start": "http://govdata.de/param/offset",
      },
    },
    ... same for Resources, Groups, Tags
  }
}
```

Mike Amundsen's “H-Factors”



Source: M. Amundsen: Hypermedia APIs with HTML5 & Node, O'Reilly, 2011

Mike Amundsen's “H-Factors”

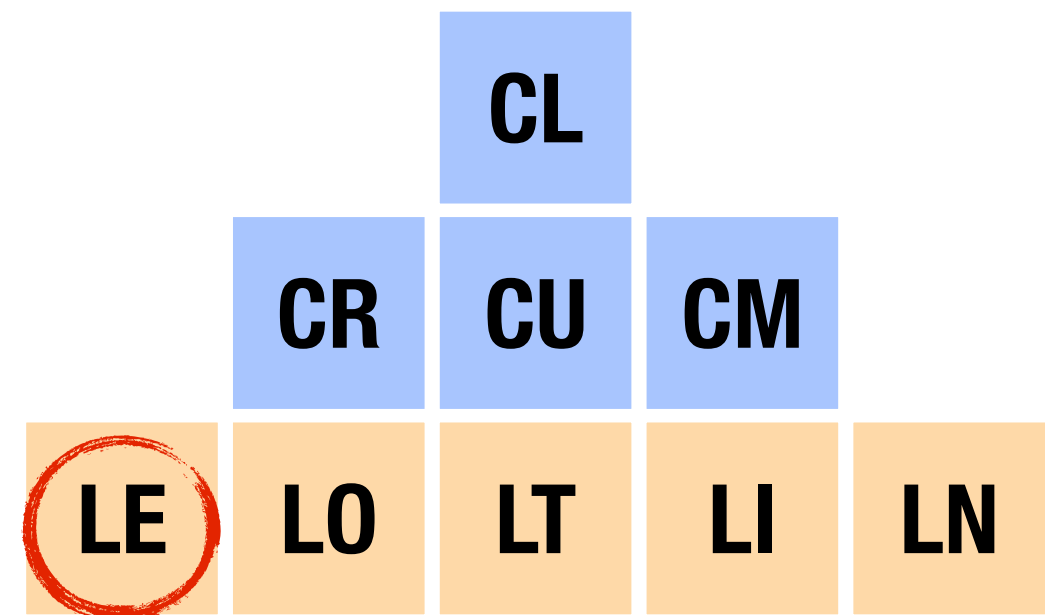
Embedding Links

- ▶ dereference URI using HTTP GET and merge result with original document content

Example: IMG markup tag in HTML

```

```



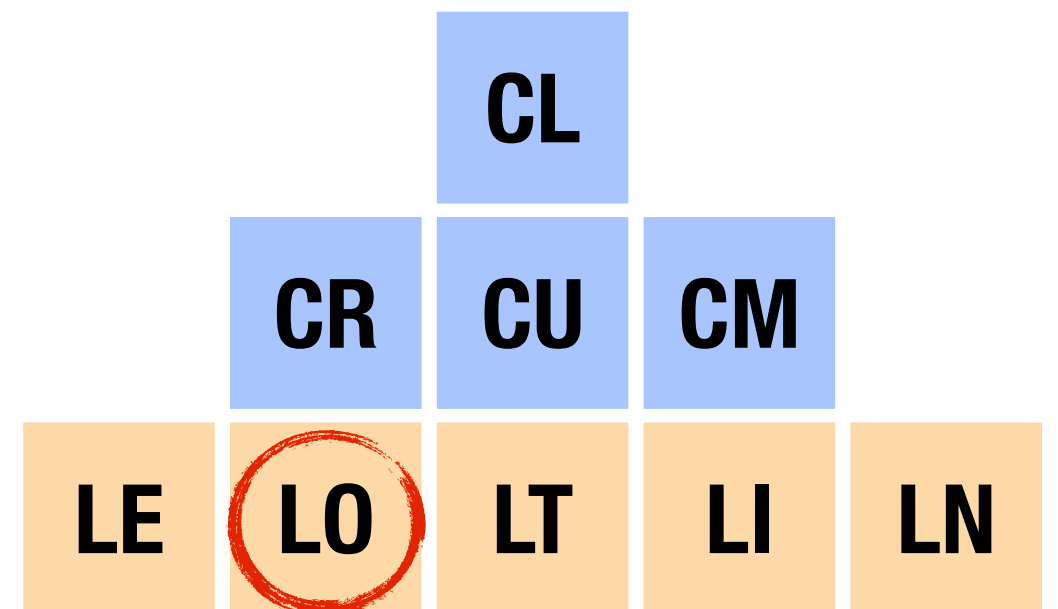
Mike Amundsen's "H-Factors"

Outbound Links

- ▶ dereference URI using HTTP GET and replace original document content with result (traversal/navigational link)

Example: A markup tag in HTML

```
<a href="..."> ... </a>
```



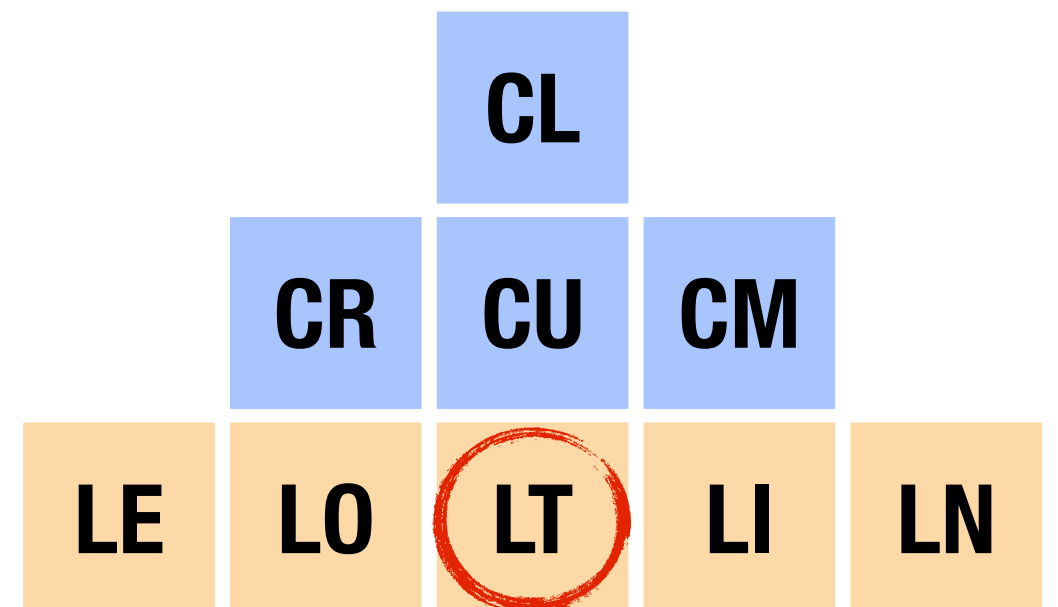
Mike Amundsen's "H-Factors"

Templated Links

- ▶ provide a way to indicate parameters that can be supplied when executing an HTTP GET

Example: URI Templates

```
<link href="http://example.org?search={search}">
```



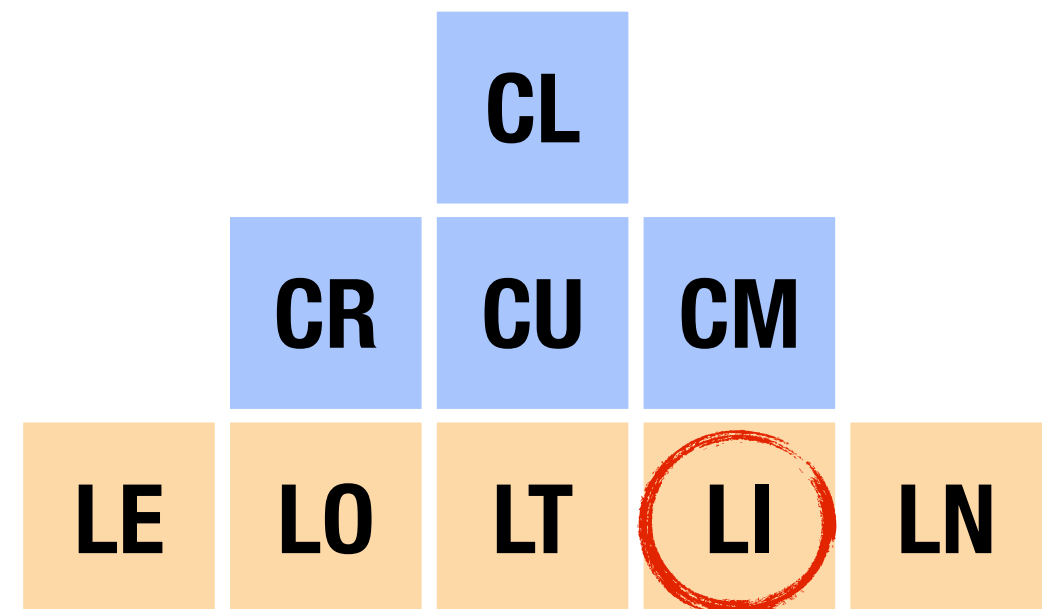
Mike Amundsen's "H-Factors"

Idempotent Links

- ▶ provides a way to define support for idempotent operations on server via HTTP PUT and DELETE methods

Example: Link with link relation type "edit" in the Atom media type

```
<link rel="edit" href="http://example.org/edit/1"/>
```



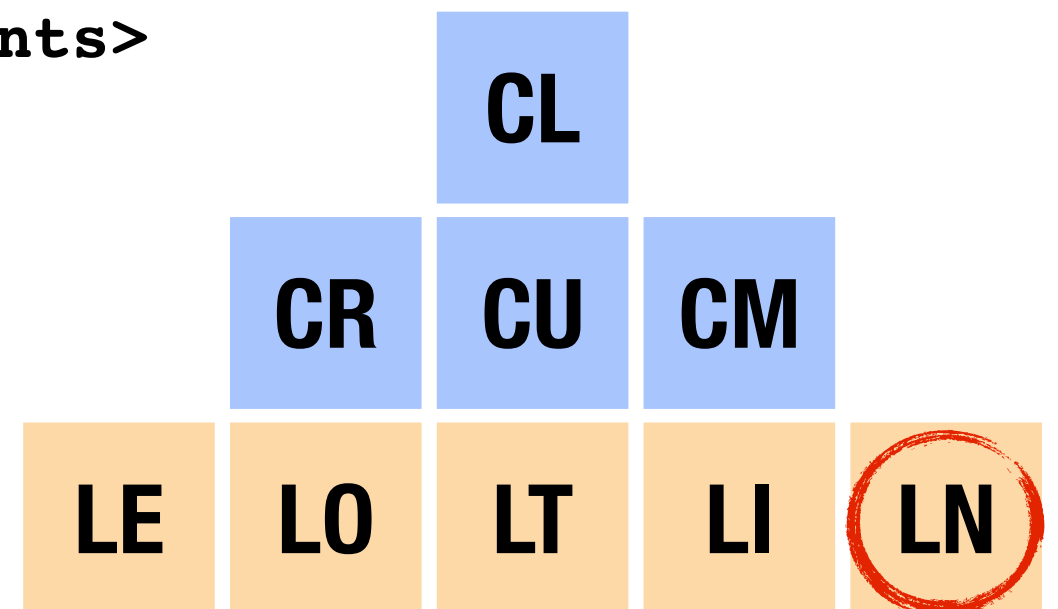
Mike Amundsen's "H-Factors"

Non-Idempotent Links

- ▶ provides a way to define support for non-idempotent operations on server via the HTTP POST method

Example: HTTP FORM element

```
<form method="post"  
    action="http://example.org/comments">  
  <textarea name="comment"></textarea>  
  <input type="submit" />  
</form>
```



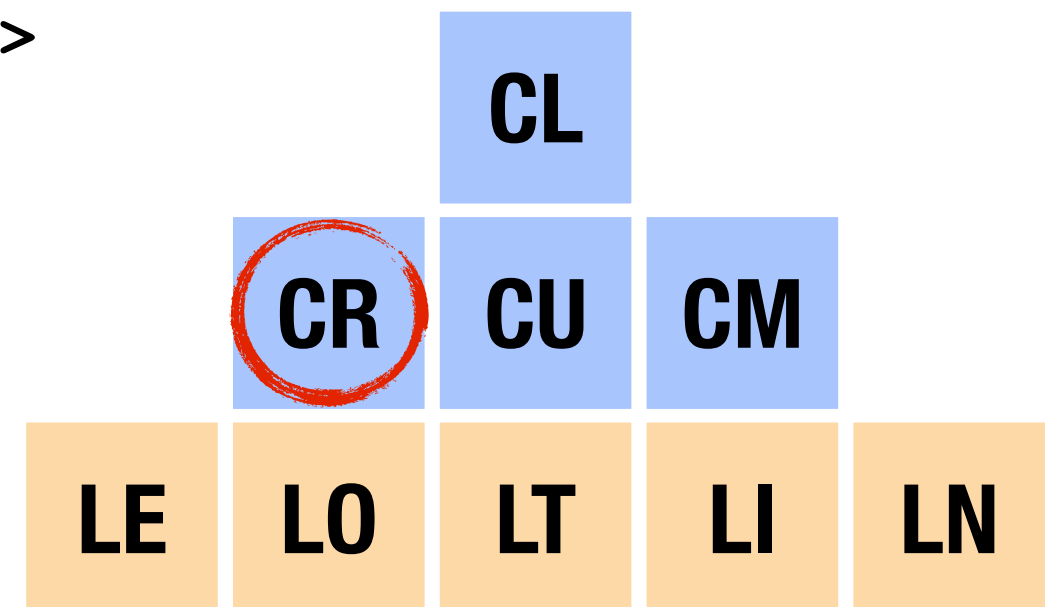
Mike Amundsen's "H-Factors"

Read Controls

- ▶ provides a way to support manipulation of control data for HTTP GET requests

Example: XInclude markup w/ accept-language attribute

```
<x:include href="http://example.org/newsfeed"  
  accept-language="de, en-gb;q=0.8" />
```



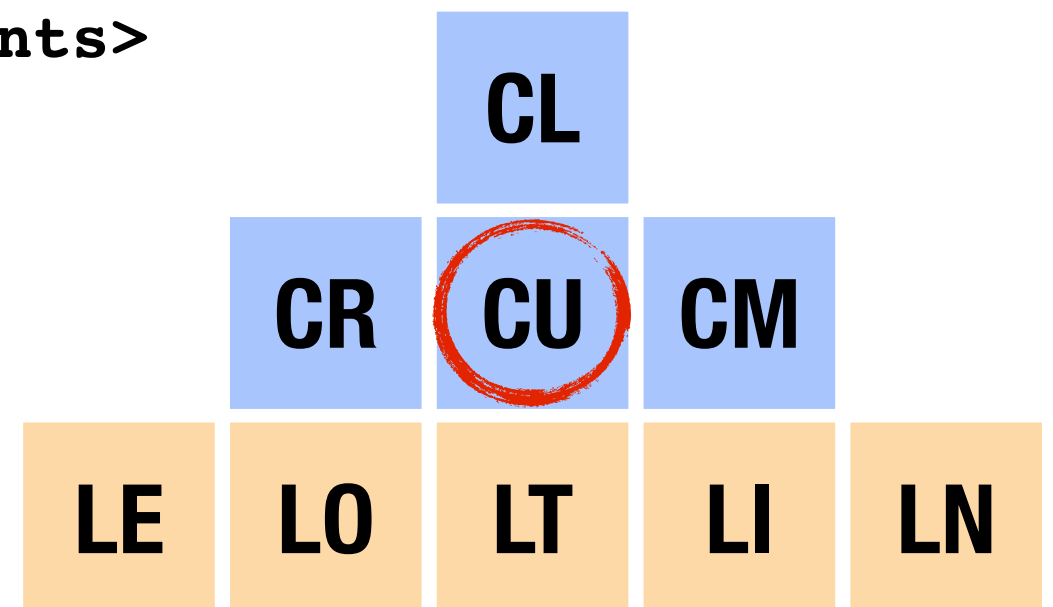
Mike Amundsen's "H-Factors"

Update Controls

- provides a way to support manipulation of control data for HTTP PUT/POST requests

Example: HTML FORM w/ enctype attribute

```
<form method="post"  
      action="http://example.org/comments"  
      enctype="text/plain"  
      <textarea name="comment"></textarea>  
      <input type="submit"/>  
</form>
```



Mike Amundsen's "H-Factors"

Method Controls

- ▶ support the ability to change the control data for the protocol method used

Example: HTML FORM's "method" element

```
<form method="post">
```

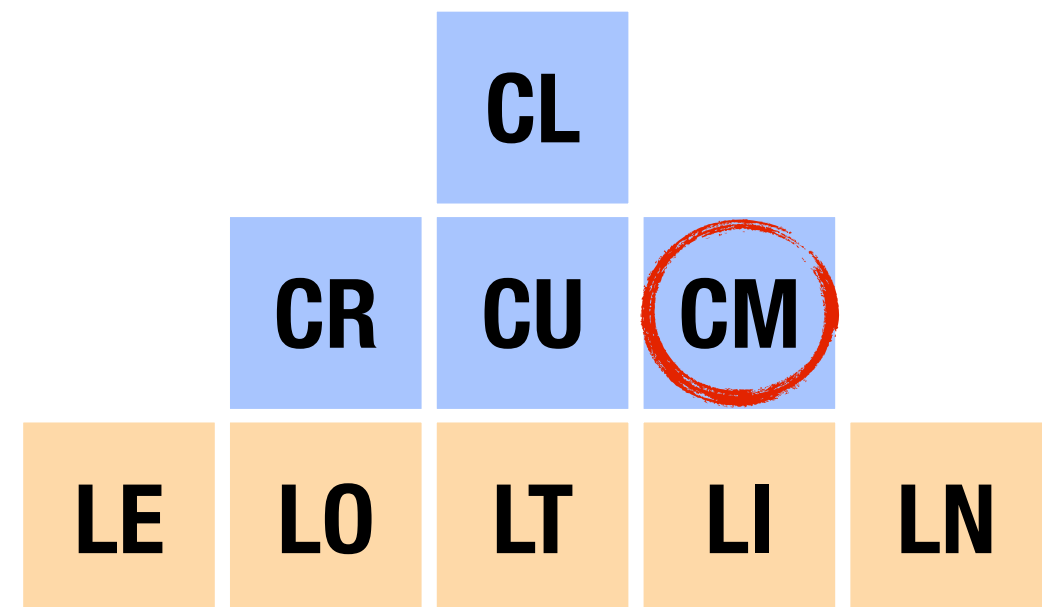
```
...
```

```
</form>
```

```
<form method="get">
```

```
...
```

```
</form>
```



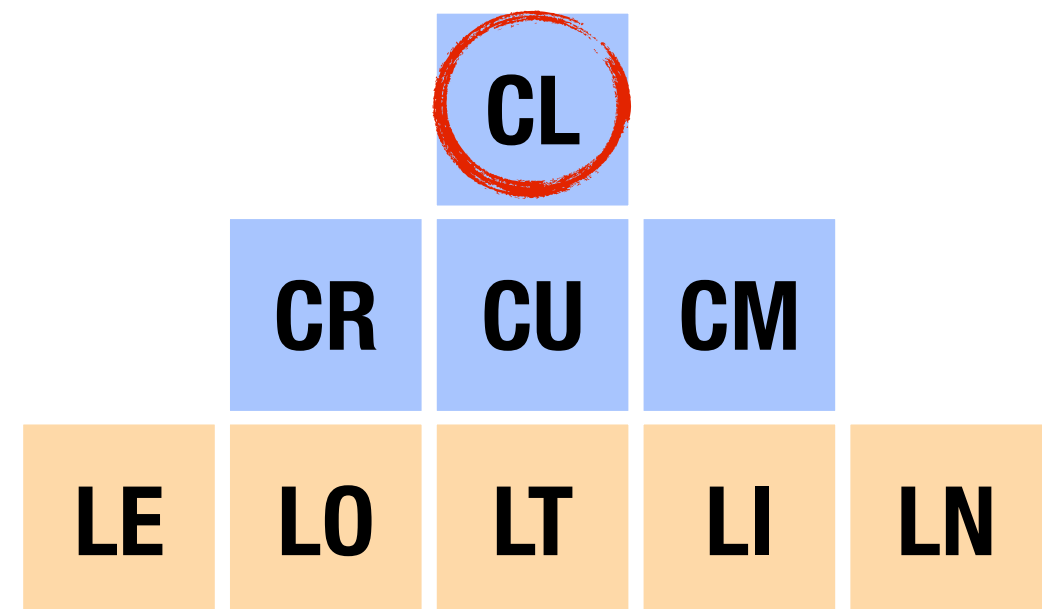
Mike Amundsen's “H-Factors”

Link Annotation Controls

- ▶ decorate links with additional metadata communicating a link's semantics to a client

Example: well-known link relation type “stylesheet” in HTML

```
<link rel="stylesheet" href="..." />
```



```
{
  "resources": {
    "http://govdata.de/rel/packages": {
      "href": "/action/package_list"
    },
    "http://govdata.de/rel/package": {
      "href-template": "/action/package_show{?id}",
      "href-vars": {
        "id": "http://govdata.de/param/package"
      },
    },
    "http://govdata.de/rel/package_search": {
      "href-template": "/action/package_search{?q,sort,rows,start}",
      "href-vars": {
        "q": "http://govdata.de/param/solr_query",
        "sort": "http://govdata.de/param/solr_sort_criteria",
        "rows": "http://govdata.de/param/max_results",
        "start": "http://govdata.de/param/offset",
      },
    },
    ... same for Resources, Groups, Tags
  }
}
```

L0

LT

LT

GET /ckan/api/action/package_list HTTP/1.1

Host: www.govdata.de

Accept: */*

HTTP/1.1 200 OK

Content-Type: application/json;charset=utf-8

```
{
  "result": [
    "destatis-dataset-81000-0108",
    "wohngebaude-und-wohnungsbestand-baden-wuerttemberg-2010",
    "destatis-dataset-12211-0114",
    "langzeitzaehlstelle_auf_der_a_62_in_weselberg",
    "langzeitzaehlstelle_auf_der_a_61_in_daxweiler",
    "destatis-dataset-46321-0006",
    "spielanlagen-hro",
    "liste-der-verkehrsunternehmen",
    "kita-stadtplan-hamburg",
    ....
  ],
  "success": true,
  "help": "Return a list of the names of the site's datasets
(packages).\n\n      :rtype: list of strings\n\n      "
}
```

L0

```
{
  "result": [
    {
      "_links": {
        "self": {"href": "/action/package_show?destatis-dataset-81000-0108" }
      }
      id: "destatis-dataset-81000-0108"
    },
    {
      "_links": {
        "self": {"href": "/action/package_show?spielanlagen-hro" }
      }
      id: "spielanlagen-hro"
    },
    ...
  ],
  "success": true,
  "help": "Return a list of the names of the site's datasets (packages).\n
\n      :rtype: list of strings\n\n      "
}
```

Wait... that's pretty chatty!
Won't that make my mobile app slow?

URI Templates to the rescue!

```
{
  "_links": {
    "items": { "href": "/action/package_show?{id}" }
  }
  "result": [
    {
      id: "destatis-dataset-81000-0108"
    },
    {
      id: "spielanlagen-hro"
    },
    ...
  ],
  "success": true,
  "help": "Return a list of the names of the site's datasets
(packages).\n\n      :rtype: list of strings\n\n      "
}
```

OK, but what about, say, adding entries?

```

{
  "_links": {
    "items": {"href": "/action/package_show?{id}" },
    "create": {"href": "/action/package_create" }
  }
  "result": [
    {
      id: "destatis-dataset-81000-0108"
    },
    {
      id: "spielanlagen-hro"
    },
    ...
  ],
  "success": true,
  "help": "Return a list of the names of the site's datasets
(packages).\n\n      :rtype: list of strings\n\n      "
}

```

LN

```
{
  "resources": {
    "http://govdata.de/rel/packages": {
      "href": "/action/package_list"
    },
    "http://govdata.de/rel/package": {
      "href-template": "/action/package_show{?id}",
      "href-vars": {
        "id": "http://govdata.de/param/package"
      },
    },
    "http://govdata.de/rel/package_search": {
      "href-template": "/action/package_search{?q,sort,rows,start}",
      "href-vars": {
        "q": "http://govdata.de/param/solr_query",
        "sort": "http://govdata.de/param/solr_sort_criteria",
        "rows": "http://govdata.de/param/max_results",
        "start": "http://govdata.de/param/offset",
      },
    },
    ... same for Resources, Groups, Tags
  }
}
```

```

{
  "resources": {
    "http://govdata.de/rel/packages": {
      "href": "/action/package_list"
    },
    "http://govdata.de/rel/package": {
      "href-template": "/action/package_show{?id}",
      "href-vars": {
        "id": "http://govdata.de/param/package"
      },
    },
    "http://govdata.de/rel/package_create": {
      "href": "/action/package_create",
      "hints": {
        "allow": ["POST"],
        "representations": ["application/json"],
      },
      ....
    }
  }
}

```

LN

CL

Conclusion

Proper usage of hypermedia elements makes Web APIs

discoverable

evolvable

robust

fun to use

Oh – one more thing...



**The CKAN API is not a
particularly well-designed Web API.**

(As far as REST principles are concerned, that is.)

Action-oriented vs. Resource-oriented

/action/package_list
/action/package_show?id={id}
/action/package_search?q={query}
...

-1

VS.

/packages
/packages/{id}
/packages?search={query}
...

+1

**Why do the Web API and the web site
have to be separate things?**

HTML5 as the media type for M2M interactions

- ▶ Properly structured “semantic” HTML works equally well for M2M interactions and for rendering in a browser

```
<ul class="packages">
  <li class="package">
    <a href="/package/destatis-dataset-81000-0108">
      destatis-dataset-81000-0108
    </a>
  </li>
  <li class="package">
    <a href="/package/spielanlagen-hro">
      spielanlagen-hro
    </a>
  </li>
  ...
</ul>
```

If JSON is preferred, there's still content negotiation

► M2M clients “see” different representation than web browsers

```
GET /packages HTTP/1.1
Host: www.govdata.de
Accept: application/json
```

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "result": [
    {
      "_links": {
        "self": {"href": "...
      },
      id: "destatis-dataset-...
    },
    ...
  ]
}
```

```
GET /packages HTTP/1.1
Host: www.govdata.de
Accept: text/html
```

```
HTTP/1.1 200 OK
Content-Type: text/html
...
<ul class="packages">
  <li class="package">
    <a href="/package/des...
      dataset-81000-0108
    </a>
  </li>
  ...
</ul>
...
```

That's it. Really.
Feel free to ask me anything!

@owolf

innoQ