



*Data Archiving and Networked Services*



# The data behind the HuisKluis

Christophe Guéret (@cgueret)

Researcher



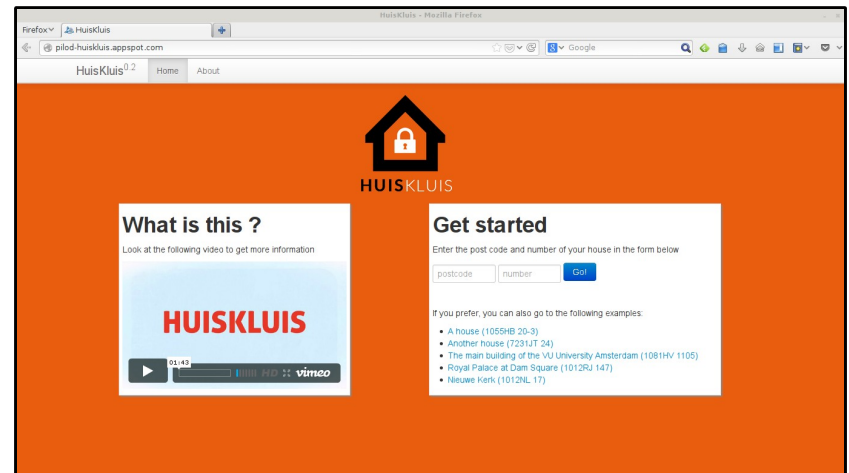
PiLOD closing event

July 3<sup>rd</sup> 2013





# Getting the open data in the app.



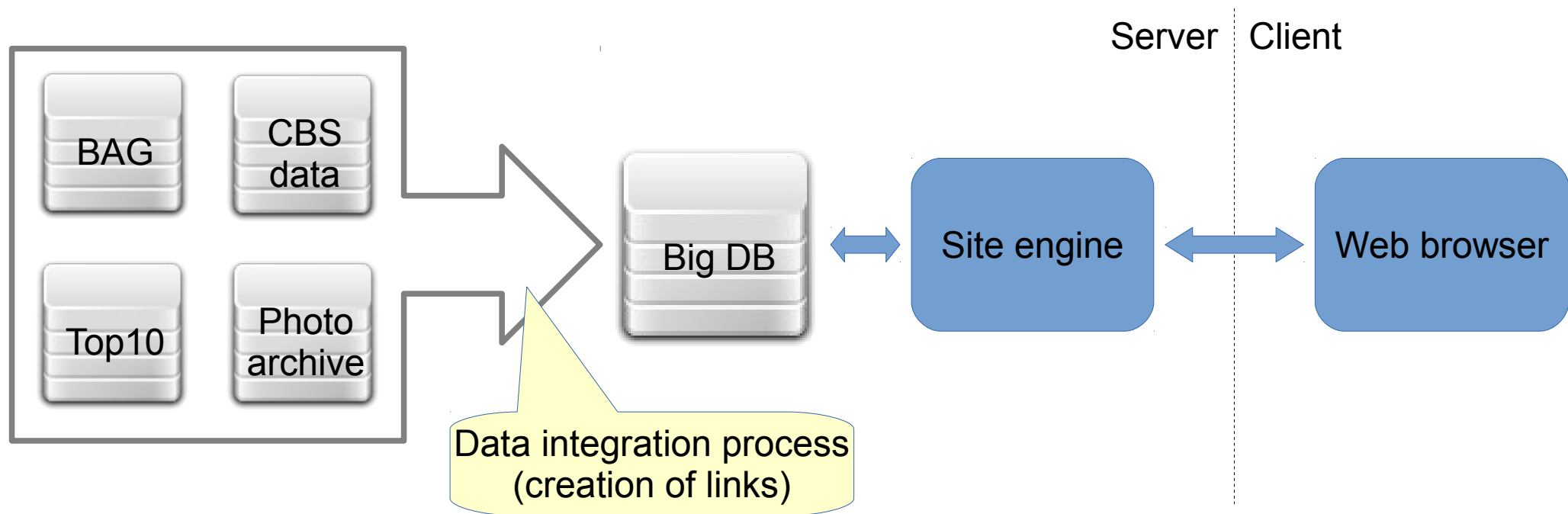
<https://data.overheid.nl/>

<http://huiskluis.nl>



# 90's style: server side black box

- Server side data store and page rendering

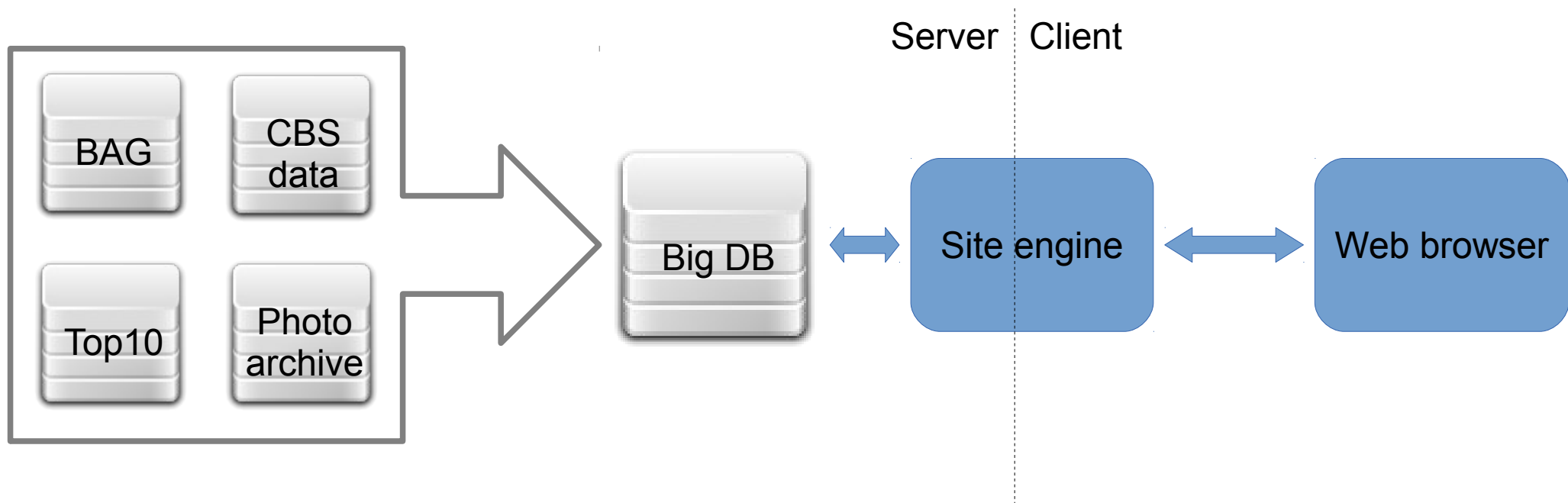


Resource intensive: server load, data integration



# Modern: Use APIs and Javascript

- Split the data from the application

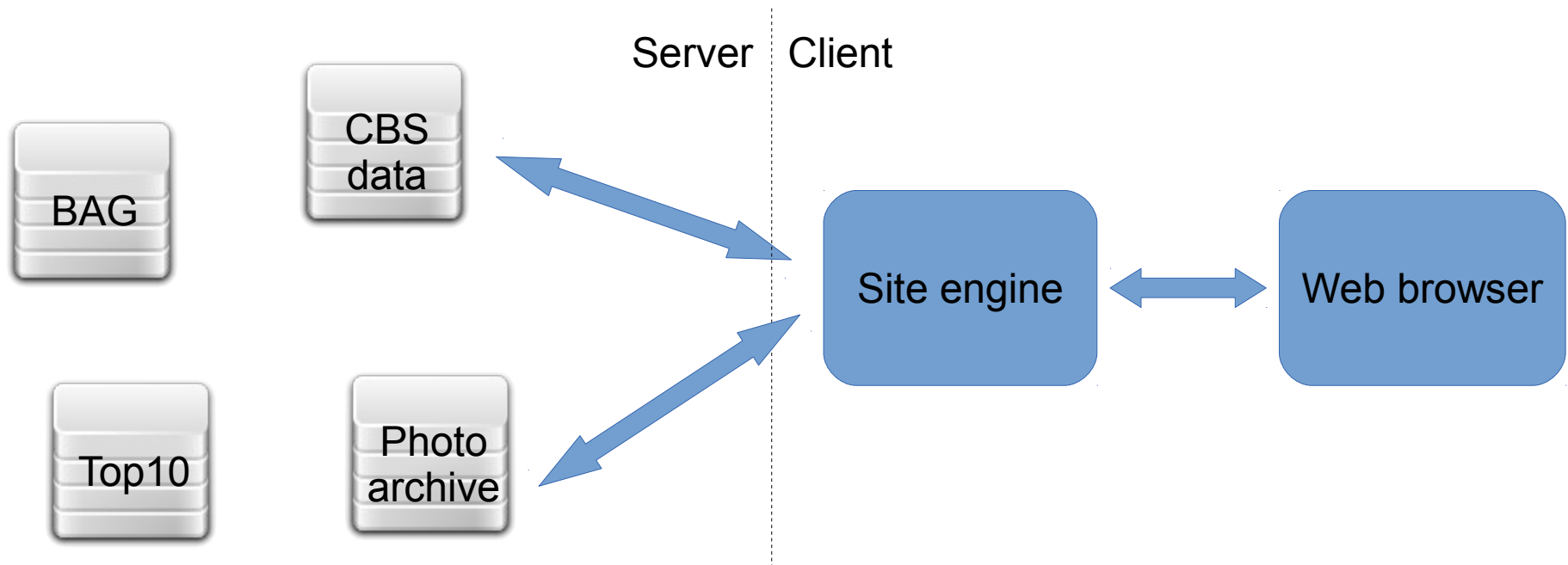




# Soon: Full Web applications



- Client side application that use only APIs





# Why APIs are so cool...

- No need to get all the data into some local store
- Application send targeted information needs
  - Leaflet: draw a map showing N 52° 22' 26" E 4° 53' 30"
  - CBS: return the statistics for around 'Gravenstraat'
  - BeeldBank: list the pictures about 'Nieuwe Kerk'
  - BAG: get the contour of the home at 1012NL-17
  - ...



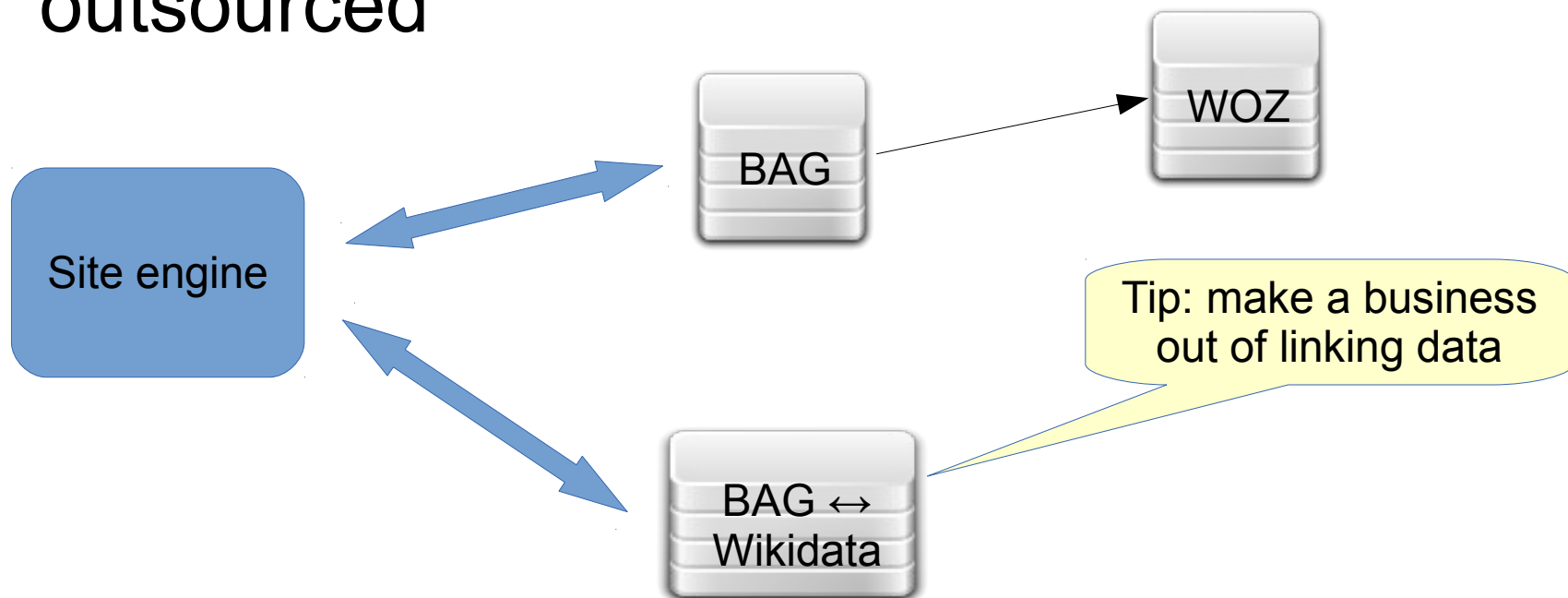
# The problem with APIs

- Some data sources do not have one
  - Provide data dumps instead
- There are few standards for APIs
  - Need to learn how to dialog with the API
- Do not tackle the data integration issue
  - Need to know what to ask



# APIs with Linked Data : links

- The links guide the client through the different APIs to find more information
- Thanks to URIs, data integration can be outsourced







# APIs with Linked Data : SPARQL

- SPARQL is a query language for RDF data
  - In practice, a *universal API* that generate *views* over the data
  - Every data need = 1 SPARQL query
- Example: HuisKluis data need from the BAG
  - “Give me the geolocation, the contour and the construction year of the building at 1012NL-17 along with the name of the street it is on”



# To use with <http://data.resc.info/bag/sparql>

PREFIX rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>

PREFIX rdfs: <<http://www.w3.org/2000/01/rdf-schema#>>

PREFIX xsd: <<http://www.w3.org/2001/XMLSchema#>>

PREFIX vocab: <<http://data.resc.info/bag/resource/vocab/>>

```
SELECT ?straat ?bouwjaar ?lat ?lng ?polygon WHERE {  
  ?huis <http://www.w3.org/ns/locn#postCode> "{POSTCODE}".  
  ?huis <http://www.w3.org/ns/locn#locatorDesignator> "{NUMMER}"^^xsd:decimal.  
  ?huis <http://www.w3.org/ns/locn#thoroughfare> ?straat.  
  ?huis <http://www.w3.org/2006/vcard/ns#longitude> ?lng.  
  ?huis <http://www.w3.org/2006/vcard/ns#latitude> ?lat.  
  ?huis vocab:adres_adresseerbaarobject ?obj.  
  ?obj vocab:verblijfsobjectactueel_pand ?pand.  
  ?pand vocab:pandactueel_bouwjaar ?bouwjaar.  
  ?pand vocab:geovlak ?polygon.  
}
```



# If you own a data set...

- Provide APIs, not only a data dump!
- Consider following the W3C standards for publishing data on the Web
  - Makes your data easier to use
  - Facilitates doing external linking





# 10 minutes overview

- Warning! Techy keywords ahead

database data LOD 5-star SPARQL  
HTML5 server-side client-side  
semantic-web linked-data client  
JSP php javascript server API rdf  
triples