Salt Run-Off Demo

Laura Kinkead
Spreadsheets will not be enough

- Working with streaming data in spreadsheets is **not feasible**
- Manipulating data across multiple spreadsheets is **not scaleable**
You should like Semantic Technology because...

- it’s a **flexible** way to access relevant portions of huge data sets
- it is easy to get that data into other **tools** (such as R)
- it helps find & analyze errors based on **metadata**
Making Scaleable Data Semantic

Data + Vocabulary = Relations + RDF + Query = Answers

- A one time encoding makes a data set forever available to intelligently interact with other data
- Queries are a flexible way to access relevant slices of data
- can query data from relations and metadata from RDF together

SELECT ?obs, ?trib, ?date
WHERE {
  ?obs raw:sample_date ?date.
}
allSaltQuery <- "
WHERE {
  ?observation raw:tributary_name ?trib_name.
  ?observation raw:sample_date ?date.
  ?observation raw:sample_acquired_time ?time.
  FILTER (?trib_name = 'Finkle Brook').
}
"
saltResults <- SPARQL(endpoint, allSaltQuery)$results

• it is easy to get semantic data into other tools (such as R)
Querying Metadata

allSaltQuery <- "
WHERE {
  ?observation raw:tributary_name ?trib_name.
  ?observation raw:sample_date ?date.
  ?observation raw:sample_acquired_time ?time.
  ?observation raw:method ?sample_method
  FILTER (?trib_name = 'Finkle Brook' && ?sample_method == 'Method A').
}"

• semantic encodings help find & analyze errors based on metadata
extra slides
where {
  ?observation raw:tributary_name ?trib_name.
  ?observation raw:sample_date ?date.
  ?observation raw:sample_acquired_time ?time.
  filter (?trib_name = "Finkle Brook" && xsd:double(?sodium) > "12"^^xsd:double && regex(?date, "-Apr-", "i").
}

(Security restrictions of this server do not allow you to retrieve remote RDF data, see details)
Results Format: HTML
Execution timeout: 0 milliseconds (values less than 1000 are ignored)
Options: Strict checking of void variables
(The result can only be sent back to browser, not saved on the server, see details)
<table>
<thead>
<tr>
<th>observation</th>
<th>trib_name</th>
<th>date</th>
<th>time</th>
<th>sodium</th>
</tr>
</thead>
</table>
Annotating the Data
Writing a Query


WHERE {
  ?observation raw:tributary_name ?trib_name.
  ?observation raw:sample_date ?date.
  ?observation raw:sample_acquired_time ?time.
  filter (?
    ?trib_name = "Finkle Brook"
    && xsd:double(?sodium) > "12"^^xsd:double
    && regex(?date, "-Apr-", "i")
  )
}

• **Flexible** way to access relevant slices of data