Six Questions (Q6) and the OpenURL Object Model (OOM)

Jeffrey A. Young
OCLC Office of Research
jyoung@oclc.org

DLF Fall Forum 2006
Boston, MA
8 November 2006
Six Questions (Q6)

- Any (web) service request can be represented by answers to the following five questions:
  1. **What** is the subject of the request?
  2. **Why** is the subject being requested?
  3. **Who** invoked the request?
  4. **Where** was the request invoked?
  5. **When**...?
    - This facet isn’t clearly modeled in Q6 or in OpenURL
- Given answers to “what”, “why”, “who”, and “where”):
  6. **How** is the information transmitted?
    1. REST or SOAP
    2. XML or Key/Value pairs
# Q6/OpenURL 1.0 Label Map

<table>
<thead>
<tr>
<th>Q6</th>
<th>OpenURL 1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. What =</td>
<td>Referent</td>
</tr>
<tr>
<td>2. Why =</td>
<td>ServiceType</td>
</tr>
<tr>
<td>3. Who =</td>
<td>Requester</td>
</tr>
<tr>
<td>4. Where =</td>
<td>ReferringEntity</td>
</tr>
<tr>
<td>5. When =</td>
<td>N/A</td>
</tr>
<tr>
<td>6. How =</td>
<td>Transport</td>
</tr>
</tbody>
</table>
Q6: “Folders” (OpenURL: “Entities”)

What: What is the subject of the request?

Why: Why is the subject being requested?

Who: Who invoked the request?

Where: Where was the request invoked?
Q6: “Clues” (OpenURL: “Descriptors”)

What
Zadie Smith’s “On Beauty”

Why
Edit review

Who
mailto:jyoung@oclc.org

Where
http://www.worldcatlibraries.org/...
Requesting a Service
Applying Q6: The “Skin” Interpretation

- Service: Edit a book review in Open WorldCat
- Q6 Breakdown:
  - How:
    - http://www.worldcatlibraries.org/wcpa/oclc/61396956;jsessionid=3D0486D8D687A2C9AE.four?page=wiki&wikitype=review
  - (BaseURL): http://www.worldcatlibraries.org/wcpa/
  - What: oclc/61396956
  - Who: ;jsessionid=3D0486D8D687A2C9AE.four
  - Why: ?page=wiki&wikitype=review
  - Where: (Taken from the HTTP ‘Referer’ header)
Why is this important?
Theory 1: Interoperability

- Standard URL patterns would make life easier for developers, **IF** those patterns are simple and intuitive
  - This might facilitate mash-ups
- Unfortunately, OpenURL has a reputation for using complex and confusing URL patterns, but those patterns are **optional**
  - New patterns can be defined by a community and added to the OpenURL Registry
Why is this important?
Theory 2: Simple Application Framework

- The OpenURL Object Model (OOM) defines language objects that mirror the abstractions in the OpenURL specification.
- Various URL patterns are treated as skins with normalized “What”, “Why”, “Who”, and “Where” folders used internally.
  - Skin 1 may expect an OCLC number “clue” to be expressed as “<oclcNo>61396956</oclcNo>” via SOAP.
  - Skin 2 may expect an OCLC number to be expressed as “oclnum=61396956” via an HTTP key/value pair.
  - Each such skin would be associated with a class that would transform this clue into the URI “info:oclnum/61396956” for internal consistency.
- Once these clues are normalized for internal consistency, adding new services should be trivial.
Questions?

- Q6 Blog
  - http://q6.oclc.org/
- OpenURL Object Model (OOM)