

# OAI Services Unbound (Prometheus or Frankenstein?)

**Jeffrey A. Young**  
**OCLC Office of Research**  
**[jyoung@oclc.org](mailto:jyoung@oclc.org)**



# The Spark of Life

- 2 Page Python Repository
- 1 Page Python Harvester
- Simple tools like these are fine for the conventional OAI model where others harvest our data and provide value-added services
- But this is our data, so what if we want to provide our own set of value-added services while we're at it?

# It's Alive!

- XML displayed in a browser is ugly
- But, you can include an XSL stylesheet reference in OAI response instructing browsers how to transform it into HTML
- The stylesheet reference is simply ignored by regular OAI harvesters
- A crude [NDLTD Union Catalog](#) interface

# The Creature Learns

- The [OpenURL Registry](#) is little more than an OAI repository plus a stylesheet.
- This registry is interesting because it contains [content](#) in addition to metadata.
- The problem, though, is that this content is buried in an OAI wrapper.
- Bright idea: extend OAI with a custom `getMetadata` verb!

# The Townsfolk Attack

- A bare-bones OAI repository can be converted into a human-friendly web application by doing a few simple tricks with OAI.
- Unfortunately, this only works for one OAI repository at a time
- Some of these tricks might be frowned on by OAI purists

# The Creature Finds Friends

## ■ OAI Registry at UIUC

- 518 repositories registered
- 329 with distinct repository identifiers
- OAI harvestable
- Many repositories still lack repository identifiers, but Tom Habing is willing to add them retrospectively

## ■ NISO asked us to create a new registry patterned after the OpenURL Registry

# The Creature Adapts

- All the tricks used on the OpenURL are based on manipulating OAI responses
- Rather than extend a particular OAI repository, these tricks can run as a separate Servlet that performs dynamic OAI requests behind the scenes to get what it needs.
- Because the URLs for these OAI extensions are so ugly, why not have this Servlet hide them behind a simple URL pattern?
- By incorporating the repository identifier into the URL pattern, they can work for anyone

# And Lives Happily Ever After

- <http://errol.oclc.org/>
- <http://errol.oclc.org/xmlregistry.oclc.org>
- <http://errol.oclc.org/xmlregistry.oclc.org.html>
- <http://errol.oclc.org/xmlregistry.oclc.org.rss>
- <http://errol.oclc.org/oai:xmlregistry.oclc.org:xoai/xoaiharvester.xsl>
- [http://errol.oclc.org/oai:xmlregistry.oclc.org:xoai/xoaiharvester.oai\\_dc](http://errol.oclc.org/oai:xmlregistry.oclc.org:xoai/xoaiharvester.oai_dc)
- <http://errol.oclc.org/oai:xmlregistry.oclc.org:xoai/xoaiharvester.xhtml>



# For more information

- See the ERRoL Project page at:
- <http://www.oclc.org/research/projects/oairesolver/>