Net::OAI::Harvester

Ed Summers
Follett Corporation

the next 45 minutes

- need to write an application that uses OAI-PMH
- want a library to make it easy to harvest metadata using OAI-PMH
- don’t have any Perl allergies
- enjoy looking at a bit of code

What is OAI-PMH?

- Open Archives Initiative Protocol for Metadata Harvesting : Carl Lagoze and Herbert Van de Sompel
- Share any kind of XML metadata over HTTP
- Repository - the provider
- Harvester - the consumer

Net::OAI::Harvester

- A OAI-PMH harvester that's both easy to use and efficient (imho).
- Object Oriented Perl
- Available on Comprehensive Perl Archive Network (CPAN)
- Other harvesting packages: OAIHarvester2 (OCLC); my.OAI (FSConsulting); oai-perl (Univ of Southampton; Perl Harvester (Virginia Tech) ; 2 page OAI.

OAI Harvesting Verbs

- Identify
- ListMetadataFormats
- ListSets
- ListIdentifiers
- GetRecord
- ListRecords

Raw OAI

- HTTP GET request
- HTTP Response containing XML
- REST: an antidote for the SOAP blues

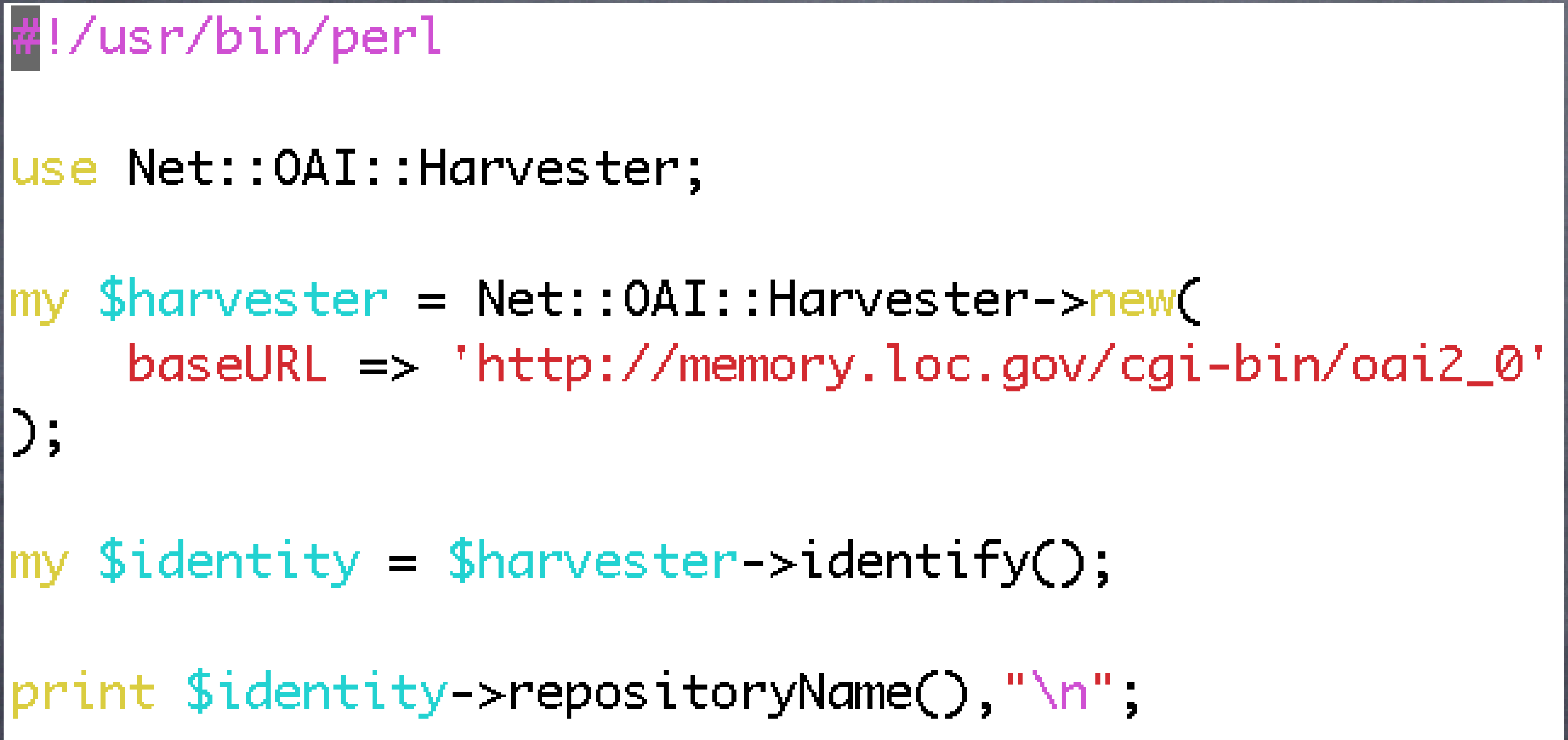
Goals

- Easy to use package for executing the 6 verbs on a given repository.
- Built in mechanisms for easily and efficiently getting at the data you want in the XML.
- Becomes a component in a larger application.

JOHN DEERE
8560

Identify

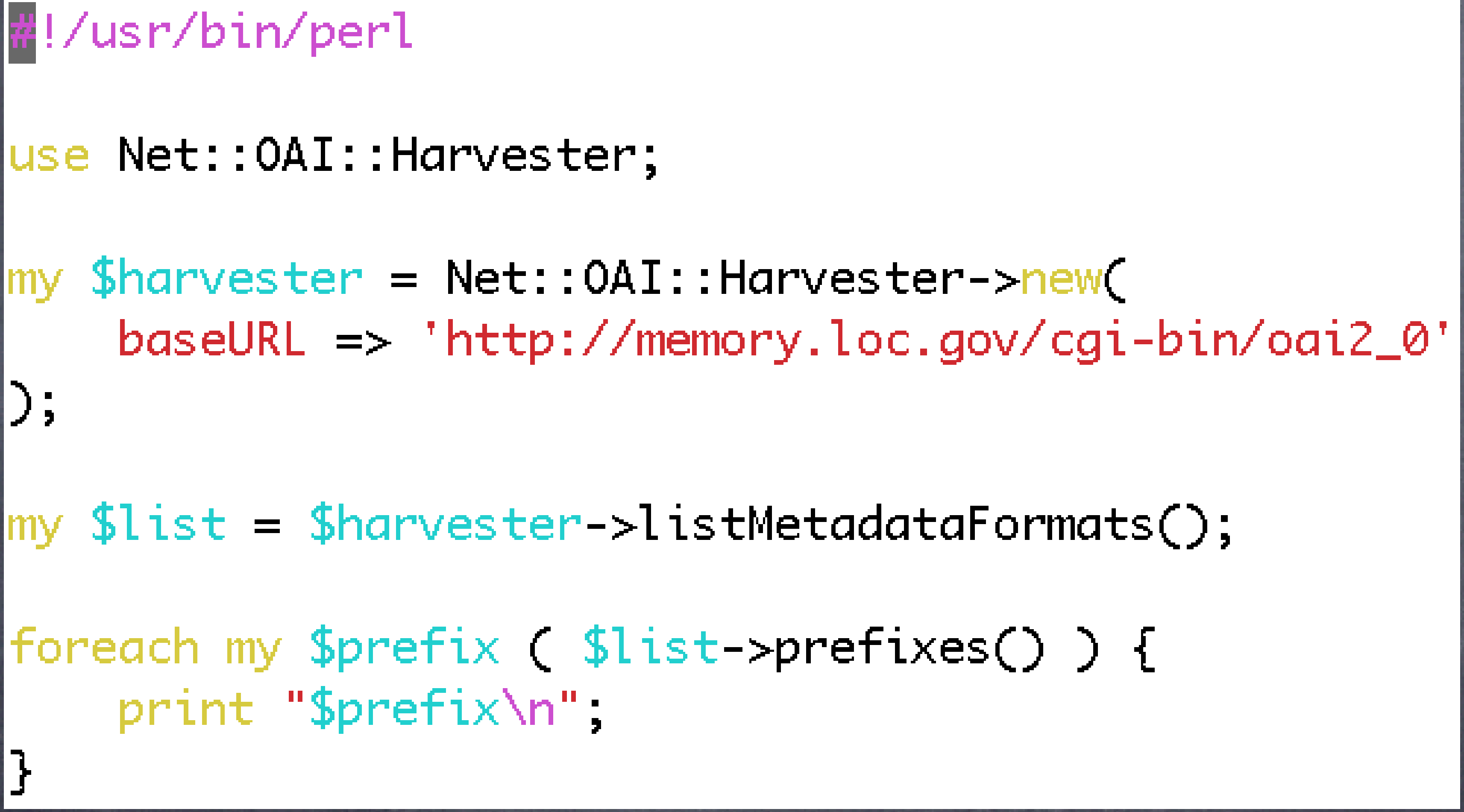
http://memory.loc.gov/cgi-bin/oai2_0?verb=Identify



```
#!/usr/bin/perl

use Net::OAI::Harvester;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $identity = $harvester->identify();

print $identity->repositoryName(),"\n";
```

ListMetadataFormats

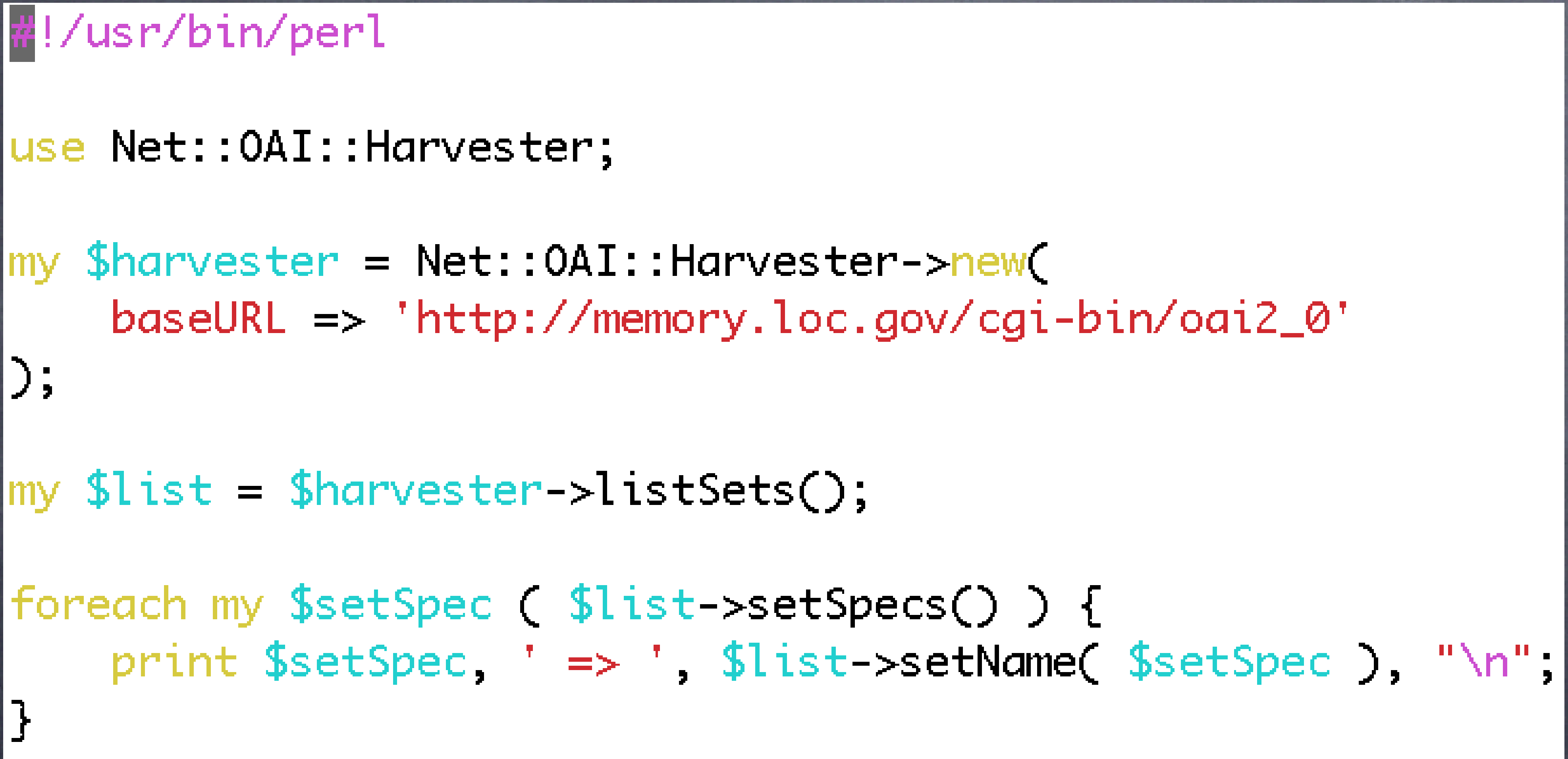
http://memory.loc.gov/cgi-bin/oai2_0?verb=ListMetadataFormats



```
#!/usr/bin/perl

use Net::OAI::Harvester;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $list = $harvester->listMetadataFormats();

foreach my $prefix ( $list->prefixes() ) {
    print "$prefix\n";
}
```

ListSets

http://memory.loc.gov/cgi-bin/oai2_0?verb=ListSets

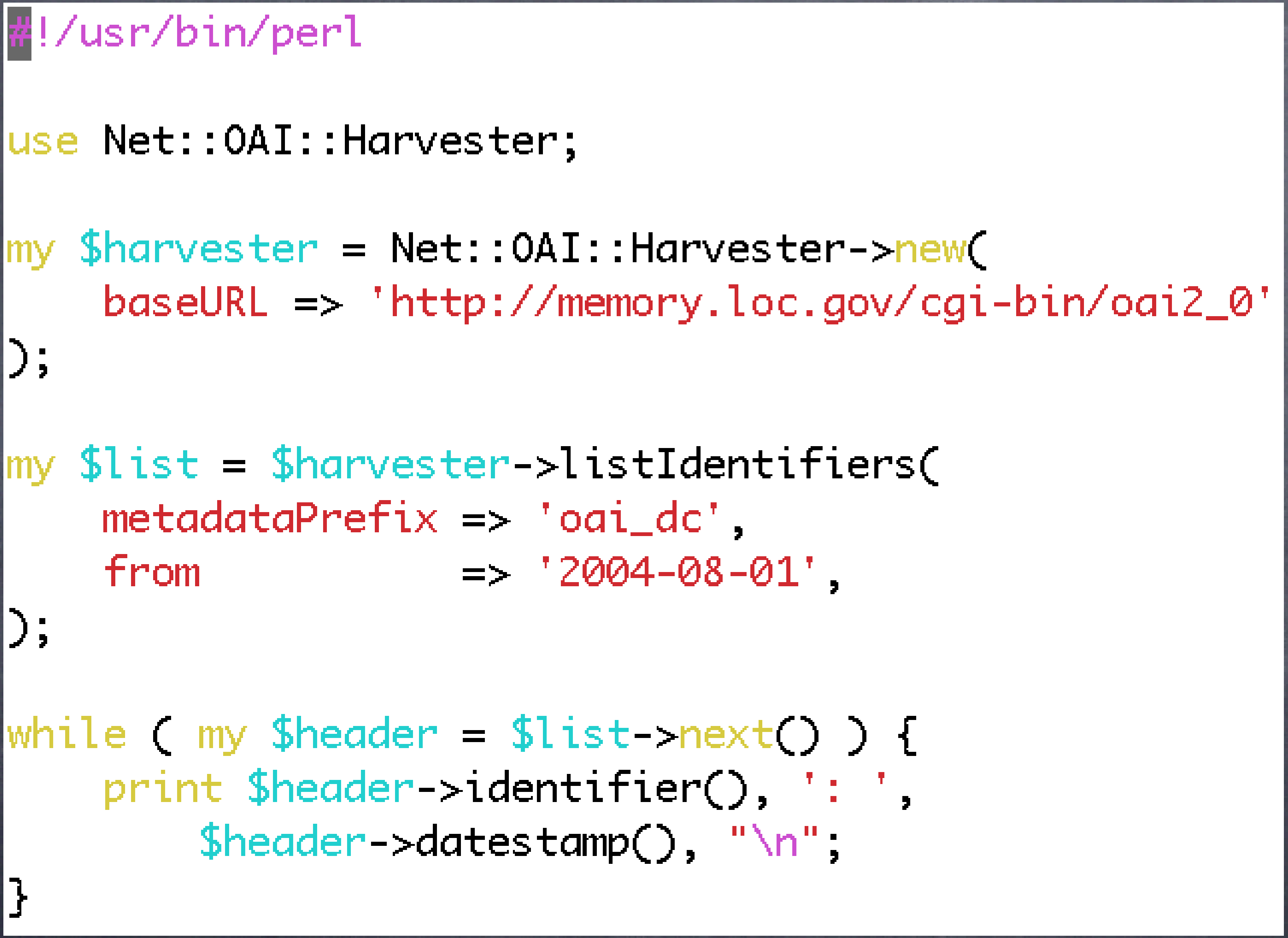


```
#!/usr/bin/perl

use Net::OAI::Harvester;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $list = $harvester->listSets();

foreach my $setSpec ( $list->setSpecs() ) {
    print $setSpec, ' => ', $list->setName( $setSpec ), "\n";
}
```

ListIdentifiers

http://memory.loc.gov/cgi-bin/oai2_0?verb=ListIdentifiers&metadataPrefix=oai_dc

http://memory.loc.gov/cgi-bin/oai2_0?verb=ListIdentifiers&metadataPrefix=oai_dc&from=2004-08-01

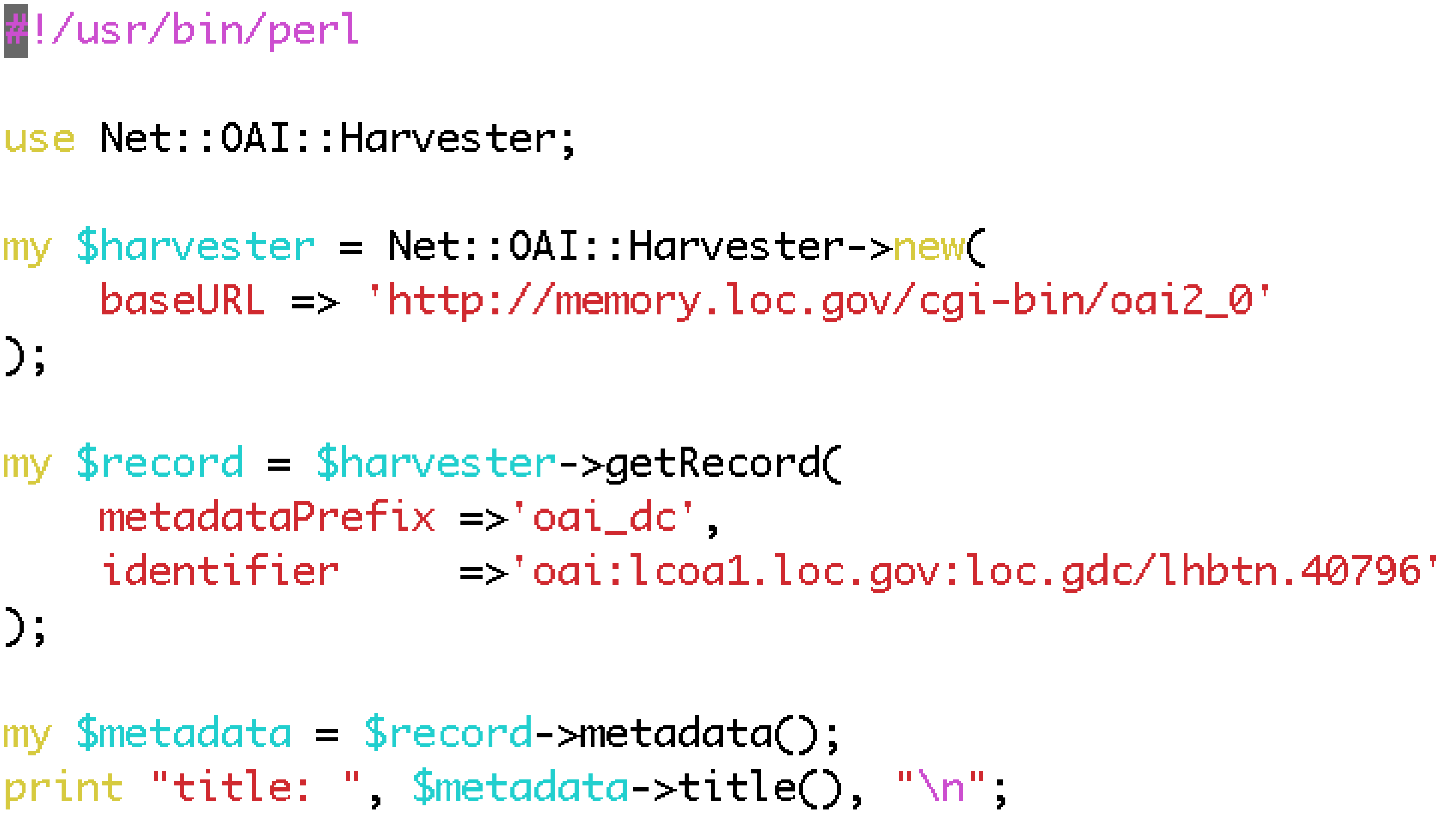


```perl
#!/usr/bin/perl

use Net::OAI::Harvester;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $list = $harvester->listIdentifiers(
    metadataPrefix => 'oai_dc',
    from           => '2004-08-01',
);

while ( my $header = $list->next() ) {
    print $header->identifier(), ': ',
        $header->datestamp(), "\n";
}
```

GetRecord

Fetch record for identifier oai:lcoa1.loc.gov:loc.gdc/lhbtn.40796

http://memory.loc.gov/cgi-bin/oai2_0?verb=GetRecord&identifier=oai%3Alcoa1.loc.gov%3Aloc.gdc%2Flhbtn.40796&metadataPrefix=oai_dc

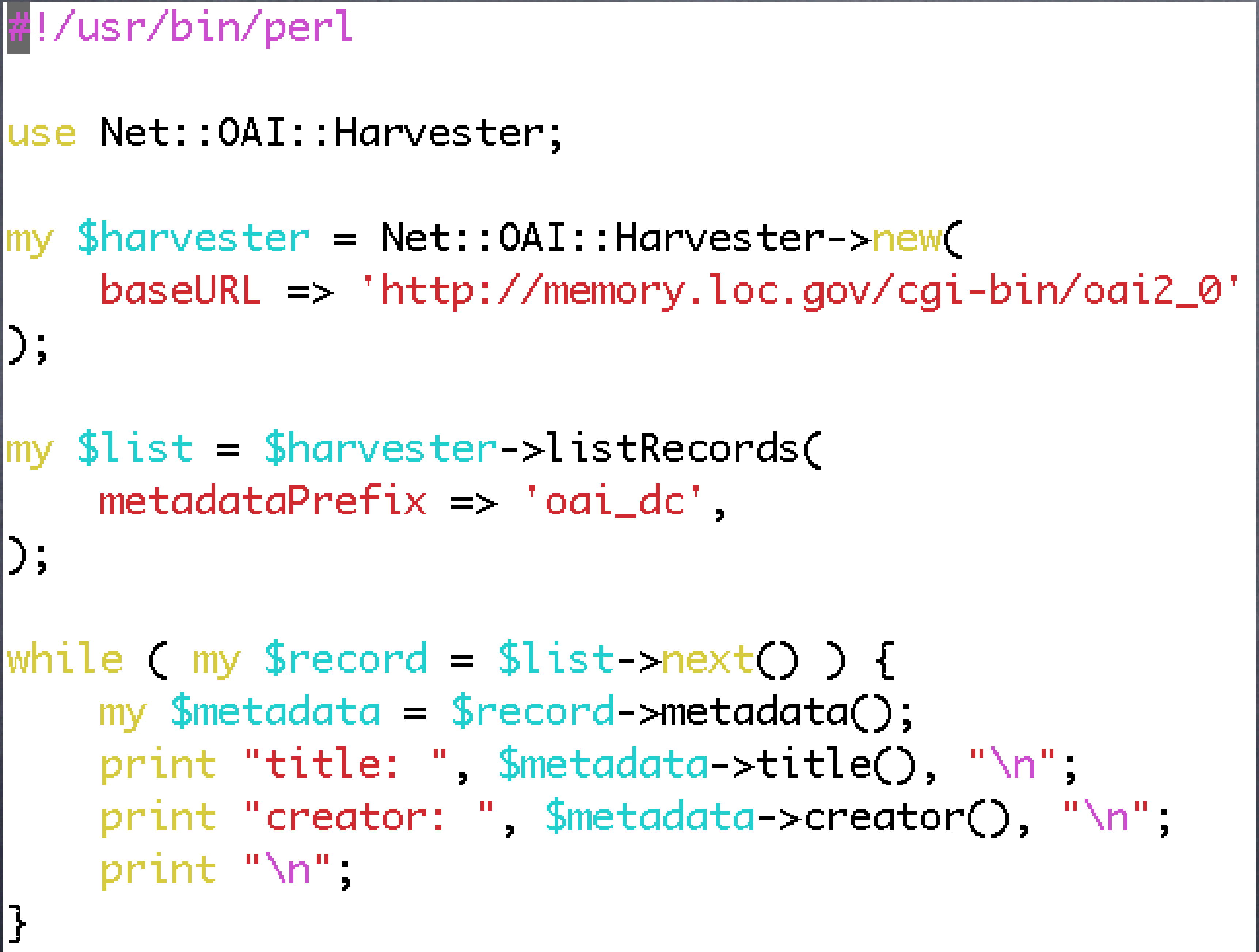


```
#!/usr/bin/perl

use Net::OAI::Harvester;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $record = $harvester->getRecord(
    metadataPrefix =>'oai_dc',
    identifier     =>'oai:lcoa1.loc.gov:loc.gdc/lhbtn.40796'
);

my $metadata = $record->metadata();
print "title: ", $metadata->title(), "\n";
```

ListRecords

http://memory.loc.gov/cgi-bin/oai2_0?verb=ListRecords&metadataPrefix=oai_dc

http://memory.loc.gov/cgi-bin/oai2_0?verb=ListRecords&metadataPrefix=oai_dc&from=2004-05-01



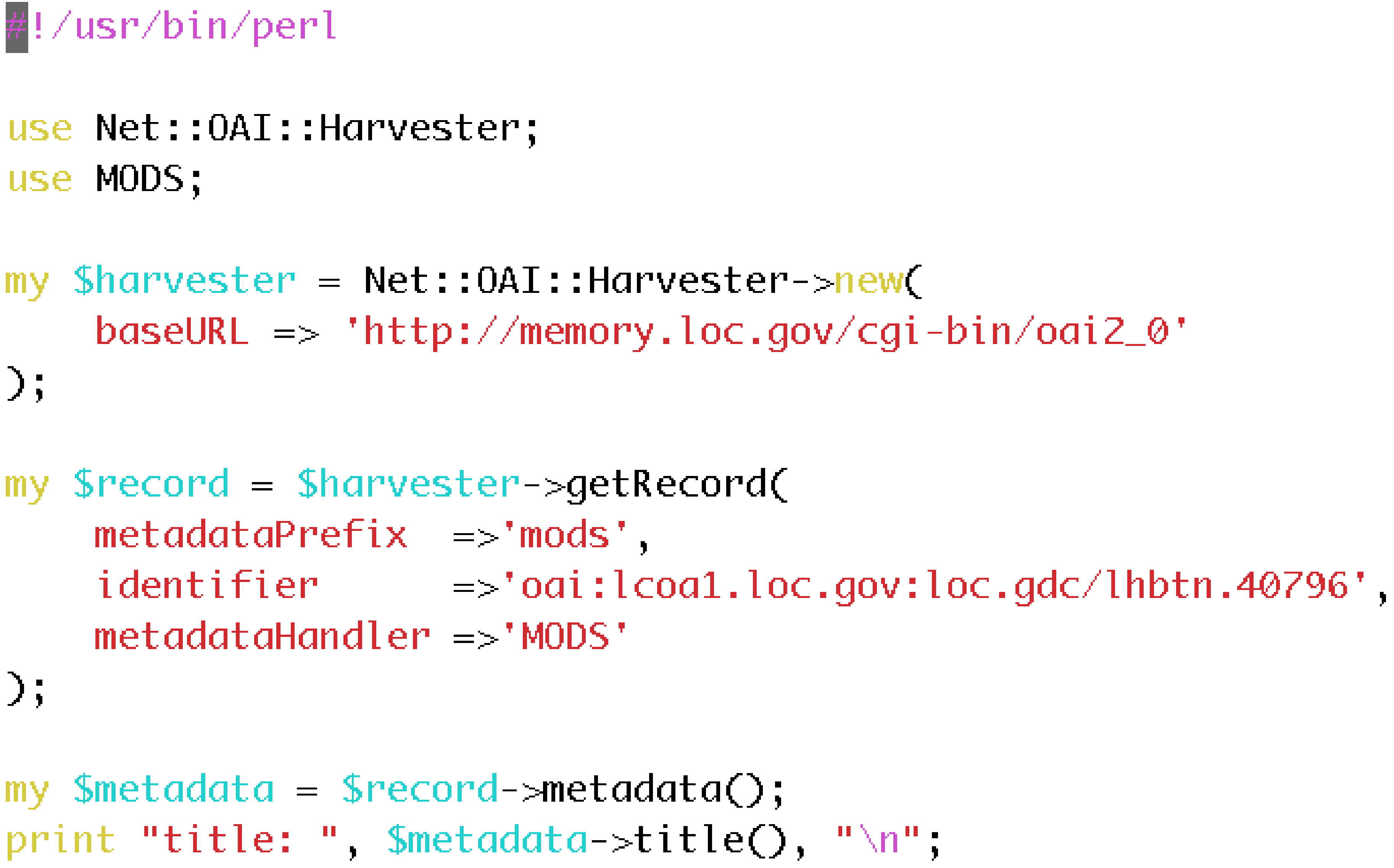
```
#!/usr/bin/perl

use Net::OAI::Harvester;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $list = $harvester->listRecords(
    metadataPrefix => 'oai_dc',
);

while ( my $record = $list->next() ) {
    my $metadata = $record->metadata();
    print "title: ", $metadata->title(), "\n";
    print "creator: ", $metadata->creator(), "\n";
    print "\n";
}
```

Resumption Tokens

- listAllRecords
- listAllIdentifiers

Only Dublin Core?

- Creating new metadata handlers for non-DC metadata: MARCXML, MODS, EAD



```perl
#!/usr/bin/perl

use Net::OAI::Harvester;
use MODS;

my $harvester = Net::OAI::Harvester->new(
    baseURL => 'http://memory.loc.gov/cgi-bin/oai2_0'
);

my $record = $harvester->getRecord(
    metadataPrefix  =>'mods',
    identifier      =>'oai:lcoa1.loc.gov:loc.gdc/lhbtn.40796',
    metadataHandler =>'MODS'
);

my $metadata = $record->metadata();
print "title: ", $metadata->title(), "\n";
```

Internals

- SAX stream based parsing: no DOM bloat
- Object serialization : not in memory
- XML Filters: easy extensibility
- Net::OAI::Base inheritance: error(), xml(), file().

Pros

- Object oriented interface which matches the OAI-PMH request methods.
- XML parsing for free.
- Error handling
- Resumption token handling

Cons

- Perl not Java, Python, etc...
- XML parsing when maybe you don’t need it.
- Faulty XML

Some Users

- Max-Planck Institute. http://www.mpg.de
- Ockham: http://www.ockham.org
- OAIster: http://oaister.umdl.umich.edu
- Emory University: http://www.emory.edu
- Journal of Chemical Education: http://chem.wisc.edu
- sdsc.edu, aps.org, u-tokyo.ac.jp, nd.edu, chem.indiana.edu, isti.cnr.it, uq.edu.au, kb.nl, osuosl.org, yu.edu, uv.es, agu.org, agrsci.dk

Installation

```
% cpan install Net::OAI::Harvester

C: ppm Net::OAI::Harvester
```

Resources

- Building OAI-PMH harvesters with Net::OAI::Harvester. http://www.ariadne.ac.uk/issue38/summers/
- Open Archives Initiative: http://www.open-archives.org
- Experimental OAI Registry at UIUC: http://gita.grainger.uiuc.edu/registry
- Perl: http://www.perl.org