POD Translation by *pod2pdf*

ajf@afco.demon.co.uk

XML.pm

Table of Contents
XML.pm

NAME

MARC::File::XML - Work with MARC data encoded as XML

SYNOPSIS

```
## reading with MARC::Batch
my $batch = MARC::Batch->new( 'XML', $filename );
my $record = $batch->next();

## or reading with MARC::File::XML explicitly
my $file = MARC::File::XML->in( $filename );
my $record = $file->next();

## serialize a single MARC::Record object as XML
print $record->as_xml();

## write a bunch of records to a file
my $file = MARC::File::XML->out( 'myfile.xml' );
$file->write( $record1 );
$file->write( $record2 );
$file->write( $record3 );
$file->close();

## instead of writing to disk, get the xml directly
my $xml = join( "\n",
    MARC::File::XML::header(),
    MARC::File::XML::record( $record1 ),
    MARC::File::XML::record( $record2 ),
    MARC::File::XML::footer()
);
```

DESCRIPTION

The MARC-XML distribution is an extension to the MARC-Record distribution for working with MARC21 data that is encoded as XML. The XML encoding used is the MARC21slim schema supplied by the Library of Congress. More information may be obtained here:
http://www.loc.gov/standards/marcxml/
You must have MARC::Record installed to use MARC::File::XML. In fact once you install the MARC-XML distribution you will most likely not use it directly, but will have an additional file format available to you when you use MARC::Batch.
This version of MARC-XML supersedes an the versions ending with 0.25 which were used with the MARC.pm framework. MARC-XML now uses MARC::Record exclusively.
If you have any questions or would like to contribute to this module please sign on to the perl4lib list. More information about perl4lib is available at *http://perl4lib.perl.org*.

METHODS

When you use MARC::File::XML your MARC::Record objects will have two new additional methods available to them:

as_xml()

Returns a MARC::Record object serialized in XML.

```
print $record->as_xml();
```

new_from_xml()

If you have a chunk of XML and you want a record object for it you can use this method to generate a MARC::Record object.

```
my $record = MARC::Record->new_from_xml( $xml );
```

Note: only works for single record XML chunks.

If you want to write records as XML to a file you can use out() with write() to serialize more than one record as XML.

out()

A constructor for creating a MARC::File::XML object that can write XML to a file. You must pass in the name of a file to write XML to.

```
my $file = MARC::XML::File->out( $filename );
```

write()

Used in tandem with out() to write records to a file.

```
my $file = MARC::File::XML->out( $filename );
$file->write( $record1 );
$file->write( $record2 );
```

close()

When writing records to disk the filehandle is automatically closed when you the MARC::File::XML object goes out of scope. If you want to close it explicitly use the close() method.

If you want to generate batches of records as XML, but don't want to write to disk you'll have to use header(), record() and footer() to generate the different portions.

```
$xml = join( "\n",
    MARC::File::XML::header(),
    MARC::File::XML::record( $record1 ),
    MARC::File::XML::record( $record2 ),
    MARC::File::XML::record( $record3 ),
    MARC::File::XML::footer()
);
```

header()

Returns a string of XML to use as the header to your XML file.

footer()

Returns a string of XML to use at the end of your XML file.

record()

Returns a chunk of XML suitable for placement between the header and the footer.

decode()

You probably don't ever want to call this method directly. If you do you should pass in a chunk of XML as the argument.
It is normally invoked by a call to next(), see *MARC::Batch* or *MARC::File*.

encode()

You probably want to use the as_marc() method on your MARC::Record object instead of calling this directly. But if you want to you just need to pass in the MARC::Record object you wish to encode as XML, and you will be returned the XML as a scalar.

TODO

- Support for character translation using MARC::Charset.
- Support for callback filters in decode().
- Command line utilities marc2xml, etc.

SEE ALSO

http://www.loc.gov/standards/marcxml/
MARC::File::USMARC
MARC::Batch
MARC::Record

AUTHORS

- Ed Summers <ehs@pobox.com>