

# MARC-NDLSH

<index.php?MARC-NDLSH>

NDLSHの活用

## MARC::NDLSH

- 国立国会図書館は、NDLSH (国立国会図書館件名標目表)の2008年版を 学術研究用途にて提供 (手続不要) しています。
  - ところが、この提供データ、「タブ区切り」にはなっていますが、表形式でもなんでもなく、そのままではプログラムから使うことができません。
  - そこで、Perlプログラムにて楽にNDLSHのデータを使えるよう、独自のPerlモジュール"MARC::NDLSH"を作成することにしました。
  - 最終的には、一石二鳥を狙っています
1. Perlモジュールを全国の大学図書館員、図書館学研究者向けに公開
  2. 「東京大学OPAC Plus "言選Web"にNDLSHの機能を付与する
- 清田先生のほうもで「汎用シソーラス探索ライブラリ」を開発しているとのこと。逆に汎用でないほうで差別化？
  - あと、SKOSも要チェック
  - 公開ページ <https://mbc.dl.itc.u-tokyo.ac.jp/MARC-NDLSH/>

### 概要 <sup>±</sup>

- NDLSHのデータは、リレーショナルDBに格納します。単にフレーズから関連語や上位下位語を取り出すには、ちょっとおおげさ (効率がよくない) ですが、次のメリットがあります。
  - i. リンク構造を表現できる
  - ii. リレーショナルモデルは使い慣れている
- Perlのインターフェイスはオブジェクト指向にします  
これによりリンクをきれいに使うことができます。たとえば、あるフレーズの上位語の関連語をとりだすには、

```
use MARC::NDLSH;
my $obj = new MARC::NDLSH;
@objs = $obj->set_word("Any Word");
foreach $entity (@objs) {
    foreach my $bt ( $entity->BT ) {
        foreach my $rt ( $bt->RT ) {
            print $rt->as_SH;
        }
    }
}
```

### NDLSHに関するメモ <sup>±</sup>

#### リレーショナルDBへの格納 <sup>±</sup>

- SQLiteにはデータ型は不要ですが、まあいちおう。

- もっとも、データ型といってもたいしたのがないのでtextばかりになってしまった...

- Redirect

- テーブル定義

フィールド名	データ型	備考
word	text	語
subdivision1	text	細目その1
subdivision2	text	細目その2
subdivision3	text	細目その3
subdivision4	text	細目その4
limitation	text	限定
id	text	件名標目のID
is_ndlsh	integer	件名標目かどうかのフラグ

- インデックス

インデックス名	フィールド
word_idx	word
subdivision41_idx	subdivision41
subdivision42_idx	subdivision42
subdivision43_idx	subdivision43
subdivision44_idx	subdivision44
limitation_idx	hoki
redirectid_idx	id

- 細目としてのみ使用する件名標目 (<< >>でくくられている)は登録しない

- Main

- テーブル定義

フィールド名	データ型	備考
id	text	件名標目のID
ndlsh	text	件名標目
SN	text	スコープノート
SA	text	参照注記
NDC9	text	NDC第9版
NDLC	text	国会図書館分類
yomi	text	件名ヨミ

- インデックス

インデックス名	フィールド
id_idx	id
ndlsh_idx	ndlsh

- LINK(リンクテーブル)

- テーブル定義

フィールド名	データ型	備考
form_id	text	
to_id	text	
relation	text	関連の種類(RT,BT,NT)

○ インデックス

インデックス名	フィールド
link_idx	from_id

● [データマッピングプログラム実装](#)

## インターフェイス仕様 <sup>+</sup>

● MARC::NDLSH

関数 (メソッド)	説明	データ型	公開可否
new	オブジェクトの生成	MARC::NDLSHオブジェクト	公開
is_SH	0(No) or 1 (Yes)	件名標目かどうか判定	公開
as_SH	文字列	件名標目形を出力	公開
as_words	文字列の配列	件名標目+UFの語を全て表示	公開
set_word	件名標目形でのリダイレクト	MARC::NDLS::Nodeオブジェクトの配列	公開

● MARC::NDLSH::Node

関数 (メソッド)	説明	データ型	公開可否
BT	上位語を求める	MARC::NDLS::Nodeオブジェクトの配列	公開
NT	下位語を求める	MARC::NDLSH::Nodeオブジェクトの配列	公開
RT	関連語を求める	MARC::NDLSH::Nodeオブジェクトの配列	公開
SN	スコープノートを出力	文字列	公開
NDC9	日本十進分類を求める	文字列	公開
NDLC	国会図書館分類を求める	文字列	公開
SA	参照注記を求める	文字列	公開
YOMI	件名のヨミを求める	文字列	公開
similar	類義語(USE and USE FOR)を求める	文字列の配列	公開

## 技術的なこと <sup>+</sup>

- Perlモジュールの作成はModule::Starterもあるが、今回はそこまでこらない

```
h2xs -AX -n MARC::NDLSH
```

- PerlのDBIモジュールを使う(例)

```

use DBI;
my $mydb = "ndlsh.db";
$dbh = DBI->connect(
    "dbi:SQLite:dbname=$mydb", "", "", {PrintError=>0}
);
if(!$hDB){
    print "$DBI::errstr ¥n";
    exit;
}

my $sth = $dbh->prepare(
    "SELECT to_id FROM table WHERE from_id=?"
);

# SQL文の実行
$sth->execute($from_id);

```

- パッケージからのDBファイル参照 [#b946e600]

```

my $NDLSH = $INC{"MARC/NDLSH.pm"};
$NDLSH = s/NDLSH.pm$/ndlsh2008/ndlsh2008.db/;

```

- 自オブジェクトの生成

## モジュールの実装 [±](#)

- [\[MARC-NDLSHモジュールの実装\]](#)をみよ

## 仕上げ [±](#)

## 検討事項 [±](#)

- 細目や区分の扱いを考えないと
  - 区分

イスラム	地理区分	イスラム
------	------	------

- 細目
 

インド--歴史--1500 ~ 1765	語	細目その1	細目その2
	インド 歴史	1500 ~ 1765	
- 補記？

インプロヴィゼーション(音楽)
-----------------

語	補記
インプロヴィゼーション	音楽

- の扱い削除します
- あと、前方一致検索とかどうする？

- 追録 (月1でできるらしい) への対応もかんがえんと
- 

Site admin: [ykoma](#)

**PukiWiki 1.4.5\_1** Copyright © 2001-2005 [PukiWiki Developers Team](#). License is [GPL](#).  
Based on "PukiWiki" 1.3 by [yu-ji](#). Powered by PHP 4.4.4-8+etch6. HTML convert time: 0.469 sec.