

# チベット語文献 KWIC 検索システムの再構築について

犬嶋 航

目 次	
1	はじめに 1
1	はじめに . . . . . 1
2	テーマ選択 . . . . . 1
3	テーマ内容 . . . . . 1
4	利用者（対象者） . . . . . 2
5	谷本さんが作られた制作物の確認 . . . . . 2
2	利用する技術について 4
1	KWIC について . . . . . 4
2	他の KWIC サイトの現状 . . . . . 6
3	Python（パイソン）について . . . . . 6
4	PHP について . . . . . 7
5	なぜ Python でなく、PHP なのか？ . . . . . 9
3	制作物について 9
1	再構築するにあたって . . . . . 9
2	必要な環境 . . . . . 10
3	システムファイル構築 . . . . . 10
4	有足字（チベット文字）について . . . . . 12
5	前後の長さ限定 . . . . . 12
6	検索終了合図アラート . . . . . 13
4	計画&作業過程 13
1	計画 . . . . . 13
2	作業過程 . . . . . 14
5	自己評価 19

## 1 はじめに

### (1) はじめに

私は卒業制作として“チベット語文献 KWIC 検索システムの再構築について”を制作を行った。これは、同ゼミで卒業された谷本さん<sup>(1)</sup>のプログラムを再構築を目標としたもので、Python から PHP へ置換を行い PHP 環境でまったく同じ KWIC システム環境を制作を行うことを目標としたものである。

### (2) テーマ選択

私が所属する福田ゼミ自体のテーマは“人の役に立つ Web サイトを制作”である。私はそれに基づいて、人の役に立つものは何なのか考えた。当初はなかなかテーマが決まらず悩んでいたところ、ゼミの指導教員である福田先生から先輩が作られたシステムの再構築はどうかと進められた。一からモノを制作するよりも最新のモノに再構築することも人の役に立つものだと考えたので最終的に私は 2009 年度に卒業された谷本さんの”チベット文献 KWIC 検索システムの構築について”から再構築といった形で論文を執筆することに決めた。

### (3) テーマ内容

2009 年度に卒業された谷本さんのプログラムは当時主に使われていた PYTHON 言語で作られていたもので、現在は PYTHON より PHP のほうが主流していることもあって今回は主に PYTHON から PHP に書き換えることを目標としながら機能を加えていくことを目標とした。

#### (4) 利用者（対象者）

ゼミ指導教員の福田先生がプロジェクトするチベット文献を対象にしたものとして作られているが、文献を入れ替えて使えばチベット文献に関わらず他の文献でも利用可能である。

#### (5) 谷本さんが作られた制作物の確認

制作する前に谷本さんが作られた制作物を確認して、KWIC 形式で表示するということはどういうことなのか、ファイル構築はどうなっているのか、どういった機能が加えられているのか確認した。

谷本さんの制作物内容について

- ・ KWIC 形式で表示を行う。
- ・ 一致語句を赤字で表示する。
- ・ 一致語句の段落番号を左側に表示させる。
- ・ 正規表現を用いた検索を可能にする。
- ・ アップロード機能の追加を行う。
- ・ 検索履歴機能の追加を行う。
- ・ ファイルの共有や譲渡を可能化にさせる。

谷本さんが作られた制作物を1つ1つの機能それぞれを触っていき、自分が感じたことから上記の箇条書きに書いてある機能から削除した機能は下記の3通りである。

- ・ アップロード機能。
- ・ 検索履歴機能の追加。
- ・ ファイルの共有や譲渡。

なぜ削除したかをここに解説していく。

- ・ アップロード機能 ファイルの共有や譲渡 アップロード機能は確

かにファイルのアップロードが可能となることで、文献の制限が大きく幅広がることできる。しかし、文献の制限がなくなるということは、検索の制限を無くすことに等しいこととなる。例えば数人がチベット文献ファイルを全てアップロードしたことを考えると、動作的に重くなって利用しづらい環境になってしまうことが考えられる。ファイルの管理が困難であり、検索入力フォームや文献一覧画面での表示方法が変わってしまう、アップロードに同名ファイルがあった場合など検索の効率を逆に下げてしまい使いづらい環境につながってしまうことが考えられる。

またオンラインシステムならまだしも、本件の制作物ではオフライン環境の中での利用になるためアップロードから利用するよりも文献ファイルから入れ替えて利用することも手段の1つであり、ローカルで利用する場合、サーバの軽減化にもなるといった利点があることが考えられるため削除した。

ファイルの共有や譲渡に関しては、上記に書いてあることに類似してしまうが、グループ内での利用者同士によるファイルの共有や交換をすることは文献の制限が無くなることに繋がる。ファイル交換をするためにグループ管理だったり、MySQL を利用する必要だったり、ウイルス対策など安全策を練る必要がある。自分の実力が足りないこともあり、今回の制作物の目標はオンラインでの利用・利用者自身が気軽に自由に検索できることを目的に置いてたため削除した。

- ・ 検索履歴機能の追加 検索履歴の機能では、何度でも見返すことが可能となって利用者の検索効率が上がることに結びつくことも可能である。履歴を残すとなると、手段が3つ挙げられる。

- クッキーを利用した方法。
- tempfile を利用した方法。
- テキストファイルを利用した方法。

それぞれの手段で試みることも考えた。しかしこの制作では特定の 1 人のために作られたものでなく、1 つのローカルサーバでもなくそれぞれのサーバで個人的に利用してもらいたい。また共有サーバとなるとプライバシー保護がされないという問題が起こりえる。事によっては 1 人によって履歴ファイルが消される可能性があることも考え、今回は検索履歴機能を削除した。

## 2 利用する技術について

### (1) KWIC について

本件における重要なポイントである。KWIC 形式で表示するということは、一致語句とその前後の文章を取り出す必要がある。(本件では前後 100 件に設定) 簡潔に言えばファイル検索するためのエンジンであり、検索効率を高めたものである。また文献の索引を作成する方法の名称であり、文献中から特定のキーワードを取り出し、前後 100 文字の文脈を取り出すといったことも可能である。

KWIC (クウィック) は KeyWord In Context の略号であり、検索結果を、KWIC(KeyWord In Context) の形式で表示を行う。また KWIC 索引は巡回索引とも呼ばれ、元々は文献の索引や目録を自動生成するために、1959 年に IBM のルーン (Luhn,H.) が考案した手法である。KWIC 索引では文章中の単語を巡回させ、全ての単語から索引を作成するものである。特定のキーワードがテキスト中のどのような文脈で用いられているかを知ることができる。また本件では、チベット文

字が登録されているテキストでどのような文脈で用いられているかを調べることが可能である。

以上からまとめたものから次のようなメリット・デメリットが考えられる。

#### KWIC のメリット・デメリットについて

##### メリット

1. 文献内から検索語句と一致したものを KWIC 形式で表示可能。
2. 日本語だけでなく、チベット語など外国語でも対応可能。
3. 利用者の要望に応じた検索が可能。
4. 知りたい情報を見つけることが可能。
5. 正規表現など曖昧な文字でも表現が出来る。
6. 一致語句が色付きで表示されわかりやすい。

##### デメリット

1. インターネット上で利用出来る KWIC 検索サイトが少ない。
2. 特定の人専用のモノに対したサイトが多いため制限が掛かってしまう。
3. 誰もが気軽に使えるとは言えない。
4. 場合によっては限られた文献でしか使えない。
5. 日本では余り知られていない。(インターネット上で使える KWIC サイトのほとんどが海外)
6. 利用目的が1つしかない。(特定のキーワードを取り出す)

## (2) 他の KWIC サイトの現状

本件と同じチベット文字の KWIC サイト、チベット文字専用の KWIC 検索が出来るサイトは指で数えられるほど少なく、利用出来たとしても、誰もが自由に使えるようなサイトはなくそれぞれのサイト専用のファイルを用いた検索でしか使えないことがわかった。ただし、2009 年当時には無かったと思われる”KWIC Finder”という PDF/Office 文書などのファイル形式に対応された Grep 検索、テキストビューアソフトウェアが開発されている。

## (3) Python (パイソン) について

オランダ人のガイド・ヴァンロッサム (Guido van Rossum) によって開発された広く使用されているプログラミング言語の一種で C++ や Java のオブジェクト指向スクリプト言語である。インデントによりブロックを表記するのが大きな特徴であり、文法がシンプルで覚えやすく、かつ奥深いものである。プラットフォームの幅広い領域において、スクリプティングや迅速なアプリケーション開発のための理想的な言語になっている。また OSI<sup>(2)</sup> に確証されたオープンソースライセンスで公開され、商用製品の開発にも無料で利用可能である。日本ではあまり普及化 (同じスクリプト言語である Ruby のが日本では使われる) してないが、海外ではよく使われる言語の 1 つとして挙っている。以上からまとめたものを次のようなメリット・デメリットが考えられる。

Python のメリット・デメリットについて

メリット

- ・ 汎用言語であり、幅広く使うことが可能。
- ・ PHP とは逆で Web に特化していない。

- ・ プログラミング言語の中で比較的シンプルで覚えやすい。
- ・ エラーを曖昧にせずにエラーメッセージを発生させてくれるため早い段階で検出可能。
- ・ PHP に負けずに追加モジュールを導入すればライブラリが豊富。
- ・ プログラムコードが読みやすい。
- ・ Web 開発にも利用可能。

#### デメリット

- ・ 国内ではマイナー。(海外ではメジャー)
- ・ Windows だとサポートされていない部分がある。
- ・ 日本語の参考書が少ないため初心者にとって難しい。
- ・ 対応レンタルサーバーが少ない。

#### (4) PHP について

PHP は、Hypertext Preprocessor を再帰的に略したものである。基本的にプログラミング言語は同じようなものであり、PHP も C 言語、Java、Perl の各言語から転用されたものがベースとして使われている。また動的 Web ページを記述することに主眼を置いて開発された言語ともいえる。PHP は最近出てきたイメージが強いが、PHP の歴史としては 20 年近くあり、今の PHP として成り立ったのがここ 15 年ほどである。特に Web 開発に適し、HTML に埋め込むことが可能なのも特徴の 1 つとして挙げられる。

HTML を出力するために多くのコマンドを記述する C 言語や Perl などの代わりに出力コードを HTML に埋め込み、PHP のコードとして開始及び、終了の処理命令として `<? ~~~ ?>` で囲まれ、これによって HTML 内での PHP モードの切り替えを行うことが可能である。つま

り、これまでの CGI<sup>(3)</sup>では外部のプログラムを命令コードによってそのたびに起動する形として取っていたが、PHP では上記の PHP モードによって PHP 実行環境によって PHP で作ったアプリを Web ブラウザーとしてわざわざ外部から呼び出さずに起動することができ、性能アップ・ロスを失くすことが可能になったのである。

最近では、有名なサイトである Yahoo や Amazon などのサイトでも採用され、開発されている状況にあり、これからより多くのサイトでも、PHP が使われる場が増えていくのもそうは遠くないとも言える。

以上からまとめたものを次のようなメリット・デメリットが考えられる。

#### PHP のメリット・デメリットについて

##### メリット

- ・ メモ帳などテキストエディットなどでも気軽にプログラミング可能。
- ・ データベースとの連携が優秀。
- ・ 初心者から上級者まで幅広く開発できる言語。
- ・ 標準ライブラリ (関数) が豊富である。
- ・ 一般的なレンタルサーバーでも動かすことが可能。
- ・ Python 技術者と比べ PHP 技術者のほうが多い。

##### デメリット

- ・ エラー発生した場合の原因が特定しにくい。
- ・ 利用用途がほぼ WEB に限定されてしまう。
- ・ バージョンによって正常に動作しない場合がある。
- ・ Java や C 言語と比べ動作が遅い。
- ・ わかりやすい参考書が少ない。

(5) なぜ Python でなく、PHP なのか？

なぜ本件では PHP を利用するか、まず私は本学である大谷大学人文情報学科で学んできたことのほぼすべてが PHP をメインとして勉強してきたことが一番大きい。しかしそれだけでなく、PHP はテキストベース (メモ帳など無料テキストエディット) で構築できる手軽さがあり、基本的に HTML ファイルに埋め込む言語であるため簡単なテキストエディタがあれば、誰でも書くことができ、デバッグも容易に行うことができ、文法が他のプログラミング言語と比べ容易であるからである。

また上記でも述べたように CGI のようなサーバ負担も少なく、実行速度が早い。また卒業された谷本さんの時代でも PHP を利用して制作を進められていたが、当時正規表現に問題があり、ユニコードが読み込まれないといった問題が生じた。しかし現在の PHP5.6 ではそういった問題は解消され、ユニコードによる文字列操作がサポートされるようになった。以上のことから私は PHP を利用したのである。

### 3 制作物について

(1) 再構築するにあたって

PHP の知識を深めた上で谷本さんが作られた KWIC システムに使われていた Python とこれから制作するとき用いる PHP のそれぞれの言語の違いは何なのかを理解し、KWIC システムとは何かを知り 2009 年度に卒業された谷本さんが作られたシステムを崩さず、システム自体は維持しながら Python から PHP へと切り替える方針としてチベット語文献 KWIC 検索システムの再構築に取り掛かった。

## (2) 必要な環境

制作物はチベット文字を扱うため、コンピュータ環境においてチベット文字の入力、表示できることが条件である。チベット文字は 0F00-0FFF ユニコードの制定はされている。2009 年度の時点では表示環境が十分でない現状が続いたが、現在 (2015 年) では、Macintosh (Mac) では System7 以降、Windows では、Windows 98 以降のバージョンにおいて、入力、表示、チベット文字・ラテン文字相互のテキスト変換の各機能が整り、実用可能な環境が提供されている。制作物のほうでチベット文字を表示するため、ユニコードは utf-8 に設定し、フォントは谷本さんの同じ設定で “Kailasa” と “Microsoft Himalaya” でサイズが 20px に指定した。

## (3) システムファイル構築

今回使用したファイルは 4 種類である。入力画面、検索結果ページの 2 つのみで分けられ、フッターやヘッダーは無く、検索に特化したシンプルな作りになっている。以下に 4 種類のファイルの解説をしていく。

1. form.php (検索語句フォームで検索語句を入力し送信するページ) 調べたい語句 (ここではチベット語) を入力し、対象する文献を選択し送信する。
2. kensaku.php (検索結果表示ページ) 調べたい語句を対象する文献の中から取り出し前後 100 文字を KWIC 化されたものをテーブル表示で表示。
3. style.css<sup>(4)</sup> (デザインシート) 見た目を調整するためのシート。

4. files (文献) 文献を入れるファイル。(下記の通り計 15 個の文献を利用)

- ・ chu\_mig\_pa\_sde\_bdun\_gyi\_don.txt
- ・ chu\_mig\_pa\_sde\_bdun\_gyi\_don\_incomplete.txt
- ・ dbu\_ma\_bden\_gnyis\_kyi\_bsdus\_don.txt
- ・ dbu\_ma\_bden\_gnyis\_kyi\_vgrel\_pa.txt
- ・ dbu\_ma\_snang\_bavi\_vgrel\_pa.txt
- ・ gtsang\_nag\_pa\_pvin\_comm.txt
- ・ phya\_pa\_rnam\_nges\_comm.txt
- ・ rnam\_grel\_Tik\_chen1.txt
- ・ rnam\_grel\_Tik\_chen2.txt
- ・ rnam\_grel\_Tik\_chen3.
- ・ rnam\_grel\_Tik\_chen4.txt
- ・ rngog\_lo\_pvin\_comm.txt
- ・ sde\_bdun\_gyi\_rgyan\_me\_tog.txt
- ・ thar\_lam\_gsal\_byed.txt
- ・ tshad\_ma\_rigs\_gter.txt

唯一谷本さんの時に使われた文献とは別の文献を使用した。尚、files(文献)の中身を変更、または追加して索引することも可能である。

#### (4) 有足字(チベット文字)について

チベット語自体は、ブータン国内の4カ国に分断されているチベット高原(インド、ネパール、中国、ブータン)に住む600万人によって使用されている。ゾンカ語、チベット語といった国でチベット文字として用いられる。

チベット文字は普通の文字と違って、分かりやすく説明すると韓国語のように複数の文字が合成して形成される複雑な文字である。このプログラムでは、1音節が1文字としてカウントされるのではなく、1音節を形成されるために組み合わせた文字が文字数カウントされる。チベット文字に用いる Unicode(ユニコード)<sup>(5)</sup>は、範囲:00F00~0FFF から成り、音節・頭記号・記号・占星術記号・数字など数種類に分けられたものがある。

#### (5) 前後の長さ限定

星研究室の KWIC サイト<sup>(6)</sup>や、前任者の谷本さんの KWIC システムでは、一致語句の前後の長さの表示を「短い(40文字)」「標準(60文字)」「長い(80文字)」のいずれかを選択することができるように作られていた。利用者の好みによって選択もでき、システムを利用する幅が広がって利用できることのできる機能とも思えたが、今回は前任者の谷本さんの KWIC システム利用者の一意見として福田先生から100文字がちょうどいいと意見を下さったため、今回は選択を無くし、前後の長さをそれぞれ100文字ずつ表示固定するようにした。

#### (6) 検索終了合図アラート

利用者が快適にシステムを利用してもらえるように、プログラムを正常に動かせない場合は、エラー処理やエラー表示する必要がある。本件で使われてるエラー処理は1つである。それは検索結果時間オーバー警告である。本件では、検索終了したという合図として基本的なアラート（エラーメッセージ）を付けている。これは付ける必要もないと考えたが、自らが何回かプログラム確認として KWIC システムを利用しているときに検索時間オーバーだったり、検索数が多かったりと、検索が延びていつ終わったのか不明確だったからためアラートを付けて試したところ、いつ検索終わったのか分かりやすくて使いやすかったため追加した。

### 4 計画&作業過程

#### (1) 計画

谷本さんの KWIC システムの現状と改良すべき点を考慮した上で、制作物は単なる改善でなく、使いやすく、見やすい KWIC システムを制作し、実用化を目指していく。

1. 谷本さんの KWIC システムを崩さずに PHP コードで最低限利用できる域に持っていく。
2. Python から PHP に書き換えを行う。
3. ソースを見やすくまとめる。
4. 検索表示をそれぞれ区切り表示する。
5. 検索語句対策 空白、半角、チベット語以外の言語対策を行う。
6. 位置情報、文字数、長さ、パーセントを表示する。
7. それぞれの文献表示に前ページ、次ページのリンクを作り、1文献

ごとに 100 件制限を設ける。

## (2) 作業過程

サイト全体の構成をページ、システムの概観を含め解説していく。私自身 KWIC システム、Python に関しては授業で耳にした程度でそこまでどんな・どういうことに使うのか何もわからないままだったため、卒業制作を制作を進めながら、KWIC とは何なのかを理解し、谷本さんの卒業制作物を実際に 1 つずつ索引かけてどういう表示でマッチしていくのか検証を行った。その中で制作を行う中で私がやるべきことはシステム自体は変更を行わずに、プログラミング言語を変換を行うことを目標としたものである。

### 1. Python から PHP へのプログラム置換。

この作業が制作物を制作するにあたって 7 割を占めてた印象が強い。とにかく css は相違ないため、Python で作られた cgi 形式データを `**.cgi` から `***.php` へと、内容も python から php へプログラム置換し、ディレクトリ取得・表示の部分を修正を行った。修正を終えた時点でレイアウトにズレが生じたため、修正を行ったのである。

### 2. 詳細ページのプログラムを追加/1 文献ごと 100 件制限を設ける。

当初は 1 文献ごとに 100 件制限を設ける上に 1 ページ 100 件表示をし、前のページと次のページのリンクを作ることを目標としていた。しかし、今回の件では 1 文献のみだと実用化するために作ることが出来たが、数文献の検索におけるズレが生じたため廃止し、1 文献ごとに 100 件制限のみ設けることにした。検索結果表示ページ (`kensaku.php`) に谷本さんが作られた詳細ページ (`shosai.cgi`) のプログラムを追加した。今までは、テーブル表示でクリックしたら別ページ

に詳細が出る仕組みだったがそれを同一ページで表示できるようにし、テーブル表示からテキストの塊表示するようにした。これをスムーズ化するために 100 件あたりを目安に区切ることがいいため 100 件超えたら別の語句で検索しなさいとエラー出るようにした。

### 3. 文字数など情報の追加。

後々要らないと感じ、パーセントのみ表示する形になったが、当初の構想通り、全体の文字数を調べるためにファイル読み込みプログラムを制作したのである。

コード内容（文字数処理）

```
if ($found = mb_ereg_search_pos()) {
    echo "位置 : ".$found[0];
    echo "<br />";
    echo "長さ : ".$found[1];
    echo "<br />";
} else {
print "Not Found";
}*/

/*$found = mb_ereg_search_pos($kensaku);
print $found;*/

/*$all_data = file(file_name);
$all_data = file_get_contents($kensaku); //ファイルを取得
foreach ($all_data as $kekka) {
print $kekka."<br />";
```

```
}  
  
$all_mojisu = strlen($all_data);//文字数を取得  
echo mb_strlen($all_mojisu);  
  
function funcPer($per) {  
    $percent = $found_pos / $all_mojisu * 100.0;  
    $per1 = floor($percent);  
    return $per1;  
}  
  
$kensaku_start = 0;  
while (($found_pos = strpos ( $all_data , $sw , $kensaku_start )) !  
    == FALSE) {  
    echo "位置は $found_pos です。";  
    $kensaku_start = $found_pos + 1;  
}  
}
```

こういう形で文字数・長さ・テキスト名を同時に出すようにした。

5. チェックボックス処理。 全選択・全解除のチェックボックスが反映されないエラーがおこったため修正を行った。エラー詳細：チェックボックスが複数選択されてるときの処理が正常に処理されていない現状を修復。

コード内容 (チェックボックス処理)

(修正前)

```
<script type="text/javascript">
<!--
function allcheck( tf ) {
var ElementsCount = document.sampleform.elements.length;
for( i=0 ; i<ElementsCount ; i++ ) { document.sampleform.elements[i].checked
}
}

function openwin() {
window.open("seiki.cgi", "", "width=800,height=800,status=yes,
resizable=yes,scrollbars=yes");
}
// -->
</script>
```

(修正後)

```
<script type="text/javascript">
<!--
    function allon() {
```

——チベット語文献 KWIC 検索システムの再構築について——

```
len = document.form1.length;
for ( var i =0 ; i < form1.length; i++) {
document.form1.elements[i].checked = true;
}
}

function alloff() {
len = document.form1.length;
for ( var i =0 ; i < form1.length; i++) {
document.form1.elements[i].checked = false;
}
}
//-->
</script>
```

## 5 自己評価

### 自己評価

- ・ シンプルイズシンプルを目標に制作に取りかかれた。
- ・ コードは見やすいコードになった。
- ・ 1つの作業に時間を掛けすぎてしまった。
- ・ どの知識に対してでもしっかり理解すべきだった。
- ・ 案のフォローとして代理の案を用意すべきだった。
- ・ アンケート運動行うべきだった。

今回の制作において、再構築という形、シンプルイズシンプルを目標に制作に取りかかれたと思える。しかし、Python から PHP に置換する作業に時間を掛けすぎてしまった。また軽量化に目を付けながら制作してしまっただけで、いくつかの機能を廃止してしまっただけで、利用者もこれから同ゼミの後輩がいつでも修正を掛けれるようシンプルで見やすいプログラミングコードにただけに、KWIC サイトの軽量化をした上で、機能追加も可能だっただろう。次に最初に計画した 1-6 までは達成できたものの、それぞれの部分でエラーが出て使えなくなったり、やってみて出来なかったから難易度を下げて続行するなど、計画立てながら進めたと言ってしまうととにかく PHP で使えるようにするといった大まかな計画を立ててしまったため 1 個 1 個作り終えた後、何を作るかどこを書き換えるか考えることで時間をロスにしてしまったため、自分の優柔不断な部分とプログラミングに関する知識をもっと理解し、出来なかったときのための代理の案を計画を立てるときに用意しておく必要があった。

- ・ また Python から PHP に移植してみる

1. コードに関して PHP より Python のほうがシンプルで見やすかった。

2. PHP のみ触らなかった自分にとったら Python が難しく感じた。
3. PHP に関して Java に頼らずスクリプトだけでシステム処理が出来るのは便利に感じた。
4. プログラミング言語は違ってても同じ表現方法が可能だとこの手で実感する事が出来た。
5. 谷本さんのコードと比較していくと多くの部分が似ている部分が多かった。

今回は Python から PHP への移植作業が多い中、PHP のみ学んできた私にとって Python とは何なのか、どう違うのか、Python と PHP っでどこが違うのか、始まる前は不安だらけで Python と PHP は全くの別物として見ていたため、移植することがなかなか難しく感じた。しかしやり終えてみればそんなに大きな差はなく、プログラミング言語は違っててもまったく同じような表現が出来るのだと逆に感動を覚えたのである。

注

- (1) 『チベット語文献 KWIC 検索システムの構築について』（2009 年度 卒業論文 谷本宏実）
- (2) Open Source Initiative の略語
- (3) Common Gateway Interface の略語 プログラムを起動するための仕組み
- (4) Cascading Style Sheets の略語
- (5) 文字コードの言語規格
- (6) <http://star.aa.tufs.ac.jp>

文献表

PHP Manual

<http://php.net/manual/ja/index.php>

IT 用語辞典 e-Words

<http://e-words.jp/>