

研究論文

日本現存朝鮮古書データベースの国際対応化の方法

総合情報基盤センター 教授 高井 正三
takai@itc.u-toyama.ac.jp

要約

日本現存朝鮮古書に関する書誌情報は、人文学部教授の藤本幸夫氏が 32 年以上に亘って調査収集したものであり、28 項目に及ぶデータが既に 15,000 件以上、調査票に蓄積され、毎年数 100 件以上の新しいデータが追加されてきている。本研究の目的は、既にデータベース化された集部約 2,700 件のデータに新規調査データを追加し、国際対応システムとして Unicode 化、Java 化を実現し、Web 上でこの DOKB(Database of Old Korean Books)データベースの検索サービスを実現することである。また、冊子体目録を作成するための、マスター・データの双行縦書き印刷等の可能な Type Setting System TeX の Unicode 対応への改良を実現し、最終的な目標は、刻手の印譜画像を追加して、データベースを再編成し、インターネット上に WebDB サービスを公開することである。ここでは、これまで研究開発してきた DOKB の国際化対応の問題点と具体的な解決方法を提案する。

キーワード

朝鮮古書書誌データベース 古文書データベース 旧字体漢字 Unicode Java システム

1. はじめに

本研究の目標は、既に調査・蓄積された日本現存朝鮮古書の書誌情報を全世界の朝鮮本研究者に情報提供するため、旧字体漢字の入出力を中心とする、データの蓄積、検索、表示、印刷をサービスする DOKB を作成し[1]、更に冊子体目録としてマスター・データの双行、三行縦書き印刷等の方法を考案し、国際対応 WebDB システムとして DOKB データベースの検索サービスを実現することである。本研究により、日本が世界に誇ることのできる Web 型 DOKB データベースを提供することができ、公開を待ち望む世界の朝鮮及び朝鮮本研究者に対し、研究資料として書誌情報、所在情報を提供し、研究能率の向上のみならず、歴史資料の後世への伝承、教育文化の醸成に貢献できると確信する。

Web プラット・フォーム上で検索するシステムは既にいくつかを試み、旧字体漢字と朝鮮固有外字を置換表示する Web 情報検索システムを試作し[2][3][4]、試験運用中である。

WebDB サービスの国際化の対応方法には、(1)先ず国際的に使用されている文字コードである Unicode 4.1 を使用すること。(2)この Unicode をサポートする Script Engine で、かつ業界で標準的に使用されている Java システ

ム、具体的には JSP (Java Server Pages), Java Servlet 及び Web Application Framework の JSF(Myfaces), Spring, Hibernate 等を駆使して「朝鮮古書検索システム」を開発すること。(3)データベース管理システム (DBMS) は Unicode をサポートしているフリーの DBMS PostgreSQLに変更してDBを再構成すること。(4)Web サーバは Linux または Windows 2003 システム上に、Apache Web Server, Tomcat その他の必要なサブ・システムを導入して稼働し、ネットワーク攻撃からのセキュリティを十分確保すること。(5)そして最後に、検索語を入力するための効率的な Unicode 漢字入力変換システム IME を使えるようにすること。などの方法がある。

さらに、検索結果を Unicode でイメージ出力するシステムを開発し、入力したマスター・データを容易に複製されないようにすることが、本研究に科された課題の一つでもある。

本研究の特色は、かつて日本が朝鮮から持ち帰った数多くの朝鮮古書の種類とその内容、所蔵情報など、貴重な調査データを収録したデータベースを世界の研究者に公開するための研究であり、国内外に例を見ない調査・研究の成果を提供する。

2. これまでの経緯

筆者らは、平成6年度から、藤本氏が調査・収集した日本現存朝鮮古書の書誌情報を整理して、パーソナル・コンピュータのエディタを使用して、調査項目にタグを付け、常用漢字を使用しながらデータの入力を開始した。平成17年末まで入力した書誌情報は、集部のものが約3,200件に達した。調査データの整理と漢字のみのデータ入力は、旧字体漢字や朝鮮固有外字のコード入力など非能率的なもの多く、データ入力の正確さを確保し、かつ国際対応とするため、世界各国の漢籍に関する古文書データベースの漢字入力の実情を調査してきた。

平成10年度からは、書誌情報データベース蓄積・提供用旧字体及び朝鮮固有外字の整備を行い、世界各国で使用されている古文書漢字データベース用の漢字コード体系を調査した。アメリカ合衆国では EACC(East Asian Character Code)7bit, 3Bytes コードを使用してデータを蓄積し、検索端末には X-Window クライアント・ソフトウェア CJK-Xterm 端末エミュレータを使用していた[1]。中華人民共和国では GB コードが、韓国では KS コードが、台湾では Big5 コードが主として使用されている。Windows OS で Unicode がサポートされるようになって、現在の Web 検索サービスでは一般に Unicode が使用されるようになってきている。平成16年度の第26回、平成17年度の第28回の2回、国際化 Unicode 会議に出席して、Unicode の現状を調査し、その後アメリカ合衆国議会図書館のアジア部門にて、DOKB のレビューを行い、世界に公開すべき古文書データベースの必要条件などを討議することができた[3]。

筆者らはこれらの調査を基に Unicode をベースとしながら、Unicode に登録されなかった旧字体フォントを @nnnnn の形式でデータベースに蓄積し、Unicode の Private Area に欠字 Missing Character を作成、登録してきた。この欠字を WEFT(Web Embedding Fonts Tool) 手法を使い、検索結果を表示するときにフォントとして埋め込む方法を提案した[2]。その後、Script Engine として php や Java による Web

データベース検索システムの開発を行ってきた。国際対応化を目指した DOKB システムは、開発途上にあり、平成18年度中の完成を目指して、残る課題を鋭意研究・開発中である。

なお、平成16年度には、藤本氏に「科学研究費補助金(研究成果公開促進費)」を申請してもらったため、冊子体目録として、マスター・データの双行縦書き印刷等の方法を考案し、出版本の見本を作成した。平成17年度にはこの科研費が付き、京都大学学術出版会から「日本現存朝鮮本研究(集部)」として刊行するため、マスター・データの分類、印刷・校正、CD-ROM の PDF サービスと Adobe Reader による検索に向けたマニュアル作成を担当した。平成18年3月、ようやく集部の目録が全て旧字体漢字を使用して刊行された。

3. 研究の方法

3.1 目標と手順

研究の目標は、朝鮮古書データベースのうち、集部約3,200件のデータをクリーニングし、索引作成に関するシステムの開発を行い、冊子体目録と国際対応システムとして、Unicode 化、Java 化を実現し、Web 上でこの DOKB データベースの検索サービスを実現することである。そのため、以下の手順で研究開発を行う。

- (1) 既にデータベース化された集部 2,700 件のデータに、新規調査データ約 500 件を追加する。
- (2) このデータベースのデータを Unicode 化 (UTF-8/UTF-16 に変換) する。
- (3) データベース・システムを Java システムとして再構築する。
 - ・Unicode サポートを実証する。
 - ・Java システムのセキュリティが堅牢であることを実証する。
- (4) 国際対応システムとして漢字入出力インターフェースを整備する。
 - ・日本、韓国、台湾、中国及びその他海外の研究者からのアクセス・テストを行う。
 - ・海外で漢字入出力インターフェースのレビューを受ける。
- (5) Web 上でこの DOKB データベースの検索サ

ービスを公開する.

・インターネットに集部を公開する.

(6) 冊子体目録として, マスター・データの双行, 三行縦書き印刷等の方法を考案する.

・Unicode 対応の LaTeX システムの改良を行う.

(7) 刻手の印譜の画像をデータベースに組み込めるようDBシステムを再編成する.

・画像のデータベース化を実現する.

(8) 漢字データベース関係の研究会, 学会, 国際会議及び海外の研究者からレビューを受け, 評価をする.

・海外でアクセス・テストとレビューを受ける.

(9) 研究成果を報告書としてまとめ, 刊行する.

3.2 研究開発の具体的計画

(1) 既にデータベース化された集部 2,700 件のデータに, 新規調査データ約 500 件を追加し, 約 3,200 件についてデータベース化する.

(2) 現行のデータはすべて S-JIS なので, これを UTF-8/UTF-16 に変換するため, @nnnn 形式で入力された欠字 (Missing Character) 及び朝鮮固有外字は, Unicode4.1 上に定義されていないかを調べ, 変換テーブルと変換プログラムを Java で作成し, DOKB に Upload する. Unicode4.1 で定義されていない欠字や朝鮮固有外字は, フォント作成ツール「武蔵システム, 外字・TrueType フォント・エディタ TTEdit V4.60, OpenType フォント・エディタ OTEdit V3.40」を使用して, Unicode の Private Area にフォントを作成し, 登録する.

Web 上では UTF-8/UTF-16 で表示するが, 欠字及び朝鮮固有外字は Web の Plugin でダウンロードするか, WEFT ツールを使って表示するときに埋め込む方法を使う.

(3) データベース・システムで Unicode をサポートするためには, 現行の PHP Script Engine を PHP6.0 または Java Server Pages に組み替える必要があり, Unicode の使用を実証するとともに, Security 確保を検証する必要がある. ここではセキュリティが堅牢であるという Java システムとして再構築する.

(4) 国際対応システムとして漢字入出力インターフェースを整備するには, 日本, 韓国, 台湾, 中国及びその他海外の研究者から使用可能な漢字入力用の IME(Input Method Editor)を用意し, アクセス・テストを行う必要がある.

既に 2005 年 9 月, 海外で漢字入出力インターフェースのレビューを受けるために, 米国議会図書館アジア部門にてレビューを受けた. 次はその時の討論の要約である.

1. All character codes of our DOKB should be changed from S-JIS to Unicode a.s.a.p.

2. Missing character code should be submitted Unicode Consortium to request registering on Unicode Supplement Plane as possible.

3. Index of our DOKB should be created more convenient to Chinese, Korean and Taiwanese.

4. Our DB systems should reproduce and implement again in order to support Unicode and international Web service with php6.0 or Java.

5. Our DOKB should be released all over the world after we published booklet bibliography.

なお, 2005 年 2 月には, 韓国大蔵経研究所にてアクセス・テストを実施した. IME は多言語対応の MS-Office を調達し, その使い勝手を調査する. 新システムが完成すれば, LOC のアジア部門でアクセス・テストを実施してもらえることになっている.

(5) Web 上でこの DOKB の検索サービスを公開するにあたって, Web 及び DB サーバを整備する. 2006 年中期の公開を目指す.

(6) 冊子体目録として, マスター・データの双行, 三行縦書き印刷等の方法を考案する. そのために, Unicode 対応の LaTeX システムの改良を行い, 検索結果の表示やマスター・データを印刷するために Unicode 版 LaTeX を使うか, 別の方法で PDF 化するかを検討し, データを複製されないような方法を講ずる必要がある.

(7) 刻手の印譜の画像をデータベースに組み込めるようDBシステムを再編成は, 今後の課題とする.

(8) 漢字データベース関係の研究会, 学会, 国際会議及び海外の研究者からレビューを受け, 海外でアクセス・テストとレビューを受ける.

将来は Ajax を活用した検索語誘導型の検索システムに改良したいが、とりあえず各国研究者対応の IME または漢字入力方法を開発する。新規システムは、韓国高麗大学文学部、台湾中央研究院計算中心及びアメリカ合衆国議会図書館アジア部門でのレビューをお願いする。

(9) 研究成果は科学研究費（基盤研究©(2)）（課題番号:16500053）の報告書としてまとめ、刊行する予定である。

3.3 開発作業内容とスケジュール

作業内容とスケジュールを以下のように設定し、開発作業を分担した。

(1) 2005 年 4 月－9 月

- ・国際的な漢字コードの調査
- ・現行システムのレビュー、問題点と解決策
- ・漢字データ処理の問題点と解決策

(2) 10 月以降

(1) データベースの詳細設計

- ・データベース定義の詳細設計
 - ・データベースの正規化とテーブルの定義
 - ・項目の設定、属性、長さ
 - ・図形データの採り入れ定義
 - ・画像の種類とデータ属性
 - ・最大データ量（30,000 件の書誌データ）
- ・DBMS の選定
 - ・MySQL
 - ・PostgreSQL
 - ・IBM UDB (DB2)
 - ・Oracle DB
- ・OS の選定と附帯ソフトウェアの準備

(2) 入出力詳細設計

- ・データ検索に必要な画面の設計
 - ・検索入力画面（項目限定版、フル項目版）
 - ・検索結果の一覧表表示画面、個別詳細表示画面（縦型、横型、PDF 版）
 - ・検索補助画面（漢字の他、日本語ではカタカナ読みサポート）
 - ・検索結果の印刷支持画面
- ・データの管理編集に必要な画面の設計
 - ・管理者の認証画面
 - ・新規データの入力（Insert）
 - ・既存データの更新（Update）

- ・既存データの削除（Delete）

- ・全件表示画面
- ・ヘルプ表示画面

(3) 漢字入力 IME または漢字入力方法の開発

- ・韓国（MS-IME）
- ・台湾（ChangJie IME）
- ・中国（Pinyin IME）
- ・日本（ATOK, MS-IME2003）
- ・英語圏（ローマ字漢字変換）

(4) その他の開発

- ・DB 使用マニュアル（日本語、英語、他）
- ・一般古文書データベースの応用へ
- ・汎用古文書データベース検索システム

(5) データのコード変換

- ・欠字@nnnnn の Unicode4.1 対応調査
- ・S-JIS→Unicode 変換テーブルの作成
- ・変換プログラムの作成、テスト
- ・変換できない漢字や朝鮮固有外字フォントの作成
- ・UTF-8 への Encode の実施

(3) 開発作業の分担

- ・全体総括及びサーバの整備
- ・DB アクセス・ツールの開発
- ・ユーザ・インターフェースの開発
- ・漢字の Unicode 変換

4. 開発

4.1 欠字・外字処理と漢字の Unicode 変換

(1) S-JIS に無い文字＝欠字

MS が制定した S-JIS コードに定義されていない文字は入力できない。その時は旧字体を用いて入力してもよいが、JIS 第2水準までとする。「余」、「芸（ウン）」等、旧字体に変換されては困る漢字がある時は、囲み線で明記しておく。また、入力できない漢字のうち、京都大学漢字典に記載されているものは、記号「@」と「康熙字典コード番号（5桁）」を連結して入れる。また、「康熙字典コード」にない漢字は、記号「@」の後ろに、朝鮮固有外字として 60000 台からの連番で数値を入れる。この連番は、共通の外字管理表を参照して、新規のものは新しい番号を発行する[4][5][6][7]。

現在まで S-JIS 欠字は 1,079 字を登録し、コード入力を行ってきたが、Unicode に定義されていない文字は、全体の 3%程度であった。このうち大部分は Unicode の基本多言語面 BMP (Basic Multilingual Plane) に存在し、第 1 面の補助的多言語面 (Supplementary Multilingual Plane, 10000₁₆~1FFFF₁₆) にはなく、続く第 2 面の補助的表意文字面 (Supplementary Ideographic Plane, 20000₁₆~2FFFF₁₆) で定義されている文字が 20 文字ほどあった[資料 5 参照]。

(2) 朝鮮固有外字

朝鮮固有外字 (異体字を含む) 等は @60000 台から順に外字原簿に登録し、@連番コードで入力する。現在まで 114 字を登録したが、その内、7 割ほどが Unicode に存在した。

(3) ハングル

ハングルは古文書の中ではそれほど多くないので @70000 台から入力し、最終的な印刷ではハングル文字に置換する。入力は韓国版の MS-IME 2003 を使うか、アレハ・ハングルというアプリケーションを使う。現在まで 183 字を @7xxxx で登録したが、Unicode は全てのハングルの定義をしているので、問題はない。

(4) 記号類

刻手名等を表す記号または記号に似た文字は、特殊外字として @90000 台から登録し、@連番コードを入力する。現在まで 57 字を登録した。これは殆ど記号に近く、「外字・TrueType フォント エディタ」 TTEdit を使用して作成するより他の手段がない。

以上、最初から Unicode で入力すれば、殆どの欠字や朝鮮固有外字を入力することができることが解った。今後のデータの追加、更新は直接 WebDB にアクセスし、管理者画面から作業を行うようにしていきたい。

4.2 ユーザ・インターフェースの開発

User Interface は Web Browser を介して、利用者用と DB 管理者用の 2 つ画面を整備することとした。今後は、Web 画面から Unicode を直接入力できるようにして、データ管理を行ってきたい。

この Web 入出力画面は、HTML タグ、ページを表示する JSP、ページ遷移や表示処理を行う JSF (Apache Myfaces) を使って作成した。画面遷移図とサンプル入出力画面の一つを [資料 1, 2] に掲載した。実行環境は Apache である。

4.3 DB アクセスと検索システムの開発

DBMS は Unicode が使える PostgreSQL (通常は「ポストグレ・エスキューエル」と発音する) を使用して、DOKB を再構築することにした。なお、目録の出版に際して全てのデータを見直し、一大クリーニングをしたので、現在のものは 13 年来の大改訂後のデータであった。

検索システム開発は JSF (Myfaces) を使用し、DB へのアクセス・ツールの開発では、Spring Framework, O/R マッピングには Hibernate を使用した。Web コンテナは Tomcat5.5 を、Web サーバは Apache を使っている[8]。DB 及びシステムの内部コードは Unicode の符号化形式 UTF-8 を使用しているので、日本語は 3 バイト・コードとなっている。WebDB システムの Software 構成図を [資料 3] に、Java のコードのサンプルを [資料 4] に掲載する。

4.4 WebDB サーバの整備

WebDB サーバは極めて安価な SOHO 用 IBM x Series206 システムを用いた。CPU は Intel Pentium 4 3.2GHz, メモリは 512MB, HDD は 80GB×2, OS は Linux で Fedora Core 3.0 を用いている。なお、これとは別個に同じ Series の Windows 2003 サーバ、HDD が Raid1 の 160GB×2 も用意した。現在整備中である。

4.5 各国 IME による連続漢字入力方法

効率的な日本語入力方法、特に連続した漢字の入力を効率的に行う IME の登場が待たれるが、本場の台湾 (繁体字) や中国 (簡体字) では、それぞれの国の特徴ある IME が用意されている。

(1) 日本

・ATOK はジャストシステムが一太郎用に発展

させてきた日本固有の IME で、結構使いやすく、S-JIS, JIS, Unicode もサポートされている。一方、MS IME 2003 は、S-JIS と Unicode をサポートする Microsoft の IME であるが、辞書がこなれていない。

(2) 韓国

Microsoft IME 2003 が国際版の Office に付いてくる。一般的には McCune Reischauer 方式(韓国のローマ字)入力で漢字変換するのがベターようだ。

(3) 台湾(繁体字)

- ・ Microsoft New Phonetic 2002a (音声:読み):これは発音を頼りに該当する漢字を選択入力する IME である。

- ・ Microsoft New ChangJie IME (部首合成):台湾ではキーボードに漢字の部首を当てはめて、これを合成する方法で入力し、妥当な文字に変換していく IME で、スピードは抜群である。

- ・ 櫻花(繁体字+日本語+ひらがな、カタカナ等):これは台湾中央研究院計算中心で紹介された IME だが、日本語の他、ひらがな、カタカナも入力できるようで、「櫻花(さくら)輸入法」と言っている。

(4) 中国(簡体字)

Microsoft Pinyin IME は、Pinyin 方式でローマ字入力する IME で、該当する同音異義語を表示し、選択する方法である。

上記何れの IME でも漢字の入力ができる。

5. 今後課題と解決方法

筆者等が開発している国際化対応 WebDB システムは、専用サーバへ移植する準備段階に入っている。ここで、今回開発した Java システムでの今後の課題とその解決方法を以下に示す。

(1) ユーザ・インターフェース

今回はユーザ画面と管理者画面を用意することで開発を進めたが、管理者画面を作成する時間が足りなかった。ユーザと同じ照会、新規追加、照会・更新、照会・削除の画面を追加開発しなければならない。画面遷移図ができているので、後は力仕事で完成できる。

(2) フォントの作成

朝鮮固有外字、記号などを含む、全体の3%

程度の漢字、朝鮮固有外字、記号の TrueType フォントを作成しなければならない。TTEdit/OTEdit フォント作成システムで Vector Fonts として作成しなければならない。

(3) 各国 IME のテスト

多言語版 MS-Office で提供される標準 IME の十分な使用テストをしていないので、WebDB での確認テストが必要である。また、各国版の Windows OS に付いている IME を使った確認テストも実施する必要がある。

(4) Unicode 第2面定義の漢字表示

Unicode4.1 の第2面(20000₁₆~2FFFF₁₆)で定義した漢字コードのフォントが、Windows PC でサポートされていないので、これを表示する確認テストを終えていない。最近では Extension C までの拡張が定義されているようであり、これを含めたフォントを整備し、実際の表示を確認しなければならない。

(5) コード変換

同様に、現代漢字の旧字体漢字への変換については、全てを確認していないので、冊子体目録(集部)が刷り上がった時点で、改めてコード変換テーブルを作成し、コード変換する予定である。

6. まとめ

国際対応化を目指した DOKB システムはほぼ90%程度の開発を終えたが、公開用のサーバの起ち上げとデータ管理者画面の整備など、残る課題を鋭意研究・開発中で、平成18年中期の公開を目指している。

今後は、刻手(刻工ともいい、書物の木版を彫った人)の印譜画像の収録をすすめてきたが、この印譜画像を含む古文書の写真画像を検索し、画像として表示するために、Google Maps 等で使用されている Ajax を使いたい。非同期型 Java Script 技術と XML(eXtensible Markup Language: 拡張マークアップ言語=利用者が自由にタグを定義でき、文書中の文字列に意味付けができる言語構造を持っているタグ言語で、メタ・データの定義に使用される。)技術を組み合わせ、Ajax(「エイジャックス」と発音する。Asynchronous JavaScript + XML)を使用して、

古文書画像を前後左右に移動させながら、かつ拡大・縮小を自在に行うことができる、究極の画像データベース検索ツールを研究し、実際にそのツールを開発する。Ajax はまた、Web 上において CJK (Chinese Japanese Korean) 用の IME を実現させる技術も持ち合わせているので、印譜画像のメタ・データベース (タイトルや著者などデータの内容に関する情報等をいう) を検索する場合、特定の国の IME が無くても、CJK 共通のローマ字入力、漢字変換が可能となる。この検索語推測汎用 IME ツールとして、合わせて研究・開発していきたい。

謝辞

筆者等がサービスしている日本現存朝鮮古書データベースの構築に当たって支援を受けた同僚の布村紀男助教授。朝鮮古刊本総合目録の作成に当たって、入力項目や入力方法、分類方法、記法、索引の採取など、本データベースに全体に関する情報の提供を受けた人文学部の藤本教授。これらのデータ入力に献身的な努力をしてくれた越野、洲崎、葉山、木戸、竹澤の女性スタッフ。そして、Java システムの構築に向けて惜しみない時間を割いてくれた工学部知能情報工学科 3 年生の喜多啓太氏、林祐司氏、米田恭章氏の 3 氏に、特に彼らのやる気と根気、努力に感謝したい。ここに記して深く感謝の意を表する。

また、本研究は文部科学省科学研究費補助金 (基盤研究(C)(2))、課題番号: 16500053) を

受けて実施した。

参考文献

- [1] 高井正三, 藤本幸夫, 日本現存朝鮮古書データベースの作成と朝鮮固有外字フォントの作成, 富山大学総合情報処理センター広報, Vol.3, No.1, 130-139, 1999.
- [2] 高井正三, 布村 紀男, 日本現存朝鮮古書データベース・システムの構築, 学術情報処理研究, No.5, 87-90, 2001
- [3] 高井正三, 布村 紀男, 日本現存朝鮮古書データベース・システムの構築方法, 情報科学フォーラム 2003(FIT2003)論文集, D-26, 57-58, 2003
- [4] 高井正三, Unicode4.0 解説, 富山大学総合情報基盤センター広報, Vol.2, 96-104, 2005
- [5] The Unicode Standard Version 4.0, The Unicode Consortium, Addison-Wesley, 2004
- [6] Unicode 標準入門, トニー・グラハム著, 乾和志・海老塚徹訳, 関口正裕監修, ISBN4-7981-0030-7, 翔泳社, 2001
- [7] 高井正三, “古文書データベースにおける日本語データ処理の諸問題”, 学術情報処理研究, Vol.9, 105-108, 2005.
- [8] 岡本隆史, 吉田英嗣, 金子宗之, 権藤夏男著, Light Weight Java, MYCOM 毎日コミュニケーションズ, ISBN4-8399-1777-9, 2005
- [9] Ajax, JavaScript によるユーザビリティ革命, WEB+DB PRESS, Vol.27, 111-142, 2005

資料

(1) DOKB Web 画面遷移図



最初のトップ画面の設計書



画面遷移の設計書

(2) ユーザ・インターフェースの画面 (抜粋)



トップ画面



検索語入力画面



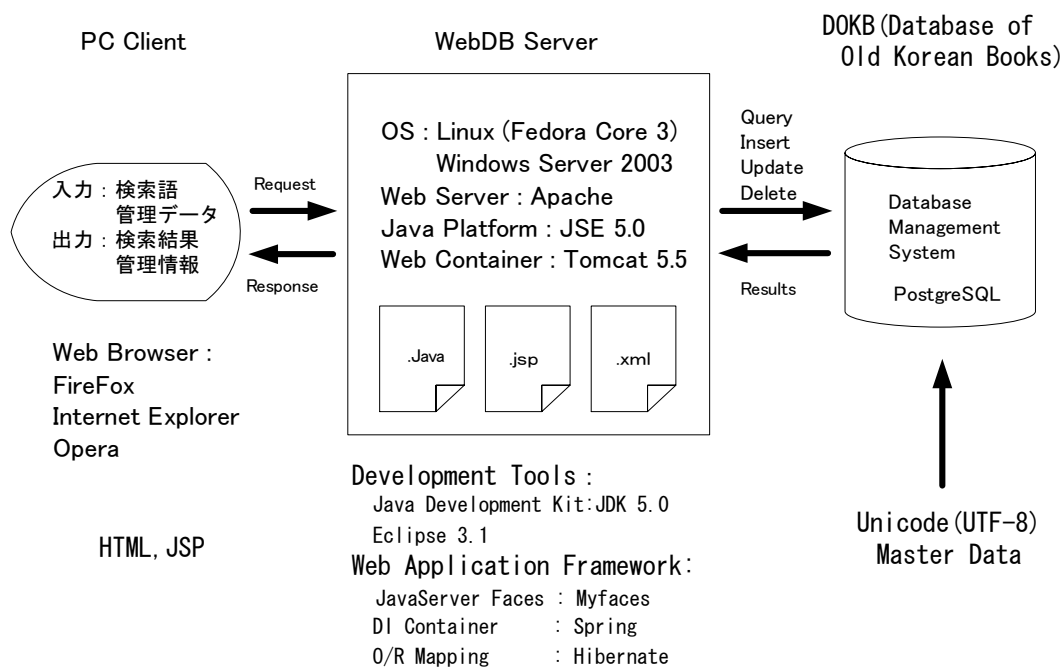
検索結果の表示画面 (一覧表示も同様)



詳細表示画面

(3) ソフトウェア構成図

Software Configuration of Web Database System named DOKB



(4) DB アクセス・ツール用 Java コードの例 (DocumentHibernateDAO.Java)

```

package jp.ac.s.tsg.acad.soc.club.dsa;

import java.util.List;

import jp.ac.s.tsg.acad.soc.club.dsa.DatabaseManager;
import jp.ac.s.tsg.acad.soc.club.dsa.DsaManager;
import jp.ac.s.tsg.acad.soc.club.dsa.DsaMember;
import jp.ac.s.tsg.acad.soc.club.dsa.DsaMemberInfo;
import jp.ac.s.tsg.acad.soc.club.dsa.DsaMemberInfoImpl;
import jp.ac.s.tsg.acad.soc.club.dsa.DsaMemberInfoImplImpl;
import jp.ac.s.tsg.acad.soc.club.dsa.DsaMemberInfoImplImplImpl;

/**
 * 主クラス
 *
 * @author 山田 太郎
 * @version 1.0
 */
public class DsaManager {
    private DatabaseManager dbm;
    private DsaManager dsm;

    /**
     * 初期化
     *
     * @param dbm データベースマネージャ
     * @param dsm DsaManager
     */
    public DsaManager(DatabaseManager dbm, DsaManager dsm) {
        this.dbm = dbm;
        this.dsm = dsm;
    }

    /**
     * データベースにデータを登録
     *
     * @param dsm DsaManager
     * @param dsmMemberInfo DsaMemberInfo
     */
    public void register(DsaManager dsm, DsaMemberInfo dsmMemberInfo) {
        dsm.register(dsmMemberInfo);
    }

    /**
     * データベースからデータを取得
     *
     * @param dsm DsaManager
     * @param dsmMemberInfo DsaMemberInfo
     * @return DsaMemberInfo
     */
    public DsaMemberInfo getDsaMemberInfo(DsaManager dsm, DsaMemberInfo dsmMemberInfo) {
        return dsm.getDsaMemberInfo(dsmMemberInfo);
    }
}

```

[illegible][illegible][illegible]

(5) S-JIS 欠字の Unicode 変換テーブル

	①00000	unicode point
漣	15771	6F36
澗	15184	6D92
淥	15305	6DE5
汶	14821	6C96
得	15252	6DC2
灑	16373	705D
濫	16312	7046
潑	15721	6F08
潏	15510	6E72
澗	51532	6F99
沔	14864	6C94
浦	15842	6F5A
潏	16274	7037

S-JIS 欠字の Unicode 対応表

字	號碼	字	號碼
鐘	60001	×	
局	60002	21C33	21C33局 11MS局
怕	60003	×	
杞	60004	233CC	10SE杞 24EDB杞
浚	60005	×	
蠱	60006	400C	
寔	60007	×	
敬	60008	×	MSI敬 450敬
弓	60009	×	3017弓
伯	60010	101C2	2MC2伯
迥	60011	迥 9008	
轎	60012	轎 6C0A	
琰	60013	×	

康熙字典にないコードの Unicode 対応表

```

1 package com.example
2
3 import java.util.Random
4
5 public class TransTable {
6
7     Random r = new Random(); // random
8
9     public TransTable() {
10         r = new Random(); // random
11     }
12
13     //各エントリは100%の確率で選択する。元の文字の出現率よりも
14     //実際の割合。もしくは変更したいとき100%の割合が100%になる
15     //程度でください。
16
17     //文字列
18     TransTable.put("a", 0.1, 0.1); // a: 10%
19     TransTable.put("b", 0.1, 0.1); // b: 10%
20     TransTable.put("c", 0.1, 0.1); // c: 10%
21     TransTable.put("d", 0.1, 0.1); // d: 10%
22     TransTable.put("e", 0.1, 0.1); // e: 10%
23     TransTable.put("f", 0.1, 0.1); // f: 10%
24     TransTable.put("g", 0.1, 0.1); // g: 10%
25     TransTable.put("h", 0.1, 0.1); // h: 10%
26     TransTable.put("i", 0.1, 0.1); // i: 10%
27     TransTable.put("j", 0.1, 0.1); // j: 10%
28     TransTable.put("k", 0.1, 0.1); // k: 10%
29     TransTable.put("l", 0.1, 0.1); // l: 10%
30     TransTable.put("m", 0.1, 0.1); // m: 10%
31     TransTable.put("n", 0.1, 0.1); // n: 10%
32     TransTable.put("o", 0.1, 0.1); // o: 10%
33     TransTable.put("p", 0.1, 0.1); // p: 10%
34     TransTable.put("q", 0.1, 0.1); // q: 10%
35     TransTable.put("r", 0.1, 0.1); // r: 10%
36     TransTable.put("s", 0.1, 0.1); // s: 10%
37     TransTable.put("t", 0.1, 0.1); // t: 10%
38     TransTable.put("u", 0.1, 0.1); // u: 10%
39     TransTable.put("v", 0.1, 0.1); // v: 10%
40     TransTable.put("w", 0.1, 0.1); // w: 10%
41     TransTable.put("x", 0.1, 0.1); // x: 10%
42     TransTable.put("y", 0.1, 0.1); // y: 10%
43     TransTable.put("z", 0.1, 0.1); // z: 10%
44     TransTable.put("A", 0.1, 0.1); // A: 10%
45     TransTable.put("B", 0.1, 0.1); // B: 10%
46     TransTable.put("C", 0.1, 0.1); // C: 10%
47     TransTable.put("D", 0.1, 0.1); // D: 10%
48     TransTable.put("E", 0.1, 0.1); // E: 10%
49     TransTable.put("F", 0.1, 0.1); // F: 10%
50     TransTable.put("G", 0.1, 0.1); // G: 10%
51     TransTable.put("H", 0.1, 0.1); // H: 10%
52     TransTable.put("I", 0.1, 0.1); // I: 10%
53     TransTable.put("J", 0.1, 0.1); // J: 10%
54     TransTable.put("K", 0.1, 0.1); // K: 10%
55     TransTable.put("L", 0.1, 0.1); // L: 10%
56     TransTable.put("M", 0.1, 0.1); // M: 10%
57     TransTable.put("N", 0.1, 0.1); // N: 10%
58     TransTable.put("O", 0.1, 0.1); // O: 10%
59     TransTable.put("P", 0.1, 0.1); // P: 10%
60     TransTable.put("Q", 0.1, 0.1); // Q: 10%
61     TransTable.put("R", 0.1, 0.1); // R: 10%
62     TransTable.put("S", 0.1, 0.1); // S: 10%
63     TransTable.put("T", 0.1, 0.1); // T: 10%
64     TransTable.put("U", 0.1, 0.1); // U: 10%
65     TransTable.put("V", 0.1, 0.1); // V: 10%
66     TransTable.put("W", 0.1, 0.1); // W: 10%
67     TransTable.put("X", 0.1, 0.1); // X: 10%
68     TransTable.put("Y", 0.1, 0.1); // Y: 10%
69     TransTable.put("Z", 0.1, 0.1); // Z: 10%
70
71     //文字列
72     TransTable.put("a", 0.1, 0.1); // a: 10%
73     TransTable.put("b", 0.1, 0.1); // b: 10%
74     TransTable.put("c", 0.1, 0.1); // c: 10%
75     TransTable.put("d", 0.1, 0.1); // d: 10%
76     TransTable.put("e", 0.1, 0.1); // e: 10%
77     TransTable.put("f", 0.1, 0.1); // f: 10%
78     TransTable.put("g", 0.1, 0.1); // g: 10%
79     TransTable.put("h", 0.1, 0.1); // h: 10%
80     TransTable.put("i", 0.1, 0.1); // i: 10%
81     TransTable.put("j", 0.1, 0.1); // j: 10%
82     TransTable.put("k", 0.1, 0.1); // k: 10%
83     TransTable.put("l", 0.1, 0.1); // l: 10%
84     TransTable.put("m", 0.1, 0.1); // m: 10%
85     TransTable.put("n", 0.1, 0.1); // n: 10%
86     TransTable.put("o", 0.1, 0.1); // o: 10%
87     TransTable.put("p", 0.1, 0.1); // p: 10%
88     TransTable.put("q", 0.1, 0.1); // q: 10%
89     TransTable.put("r", 0.1, 0.1); // r: 10%
90     TransTable.put("s", 0.1, 0.1); // s: 10%
91     TransTable.put("t", 0.1, 0.1); // t: 10%
92     TransTable.put("u", 0.1, 0.1); // u: 10%
93     TransTable.put("v", 0.1, 0.1); // v: 10%
94     TransTable.put("w", 0.1, 0.1); // w: 10%
95     TransTable.put("x", 0.1, 0.1); // x: 10%
96     TransTable.put("y", 0.1, 0.1); // y: 10%
97     TransTable.put("z", 0.1, 0.1); // z: 10%
98     TransTable.put("A", 0.1, 0.1); // A: 10%
99     TransTable.put("B", 0.1, 0.1); // B: 10%
100    TransTable.put("C", 0.1, 0.1); // C: 10%
101    TransTable.put("D", 0.1, 0.1); // D: 10%
102    TransTable.put("E", 0.1, 0.1); // E: 10%
103    TransTable.put("F", 0.1, 0.1); // F: 10%
104    TransTable.put("G", 0.1, 0.1); // G: 10%
105    TransTable.put("H", 0.1, 0.1); // H: 10%
106    TransTable.put("I", 0.1, 0.1); // I: 10%
107    TransTable.put("J", 0.1, 0.1); // J: 10%
108    TransTable.put("K", 0.1, 0.1); // K: 10%
109    TransTable.put("L", 0.1, 0.1); // L: 10%
110    TransTable.put("M", 0.1, 0.1); // M: 10%
111    TransTable.put("N", 0.1, 0.1); // N: 10%
112    TransTable.put("O", 0.1, 0.1); // O: 10%
113    TransTable.put("P", 0.1, 0.1); // P: 10%
114    TransTable.put("Q", 0.1, 0.1); // Q: 10%
115    TransTable.put("R", 0.1, 0.1); // R: 10%
116    TransTable.put("S", 0.1, 0.1); // S: 10%
117    TransTable.put("T", 0.1, 0.1); // T: 10%
118    TransTable.put("U", 0.1, 0.1); // U: 10%
119    TransTable.put("V", 0.1, 0.1); // V: 10%
120    TransTable.put("W", 0.1, 0.1); // W: 10%
121    TransTable.put("X", 0.1, 0.1); // X: 10%
122    TransTable.put("Y", 0.1, 0.1); // Y: 10%
123    TransTable.put("Z", 0.1, 0.1); // Z: 10%
124
125    //文字列
126    TransTable.put("a", 0.1, 0.1); // a: 10%
127    TransTable.put("b", 0.1, 0.1); // b: 10%
128    TransTable.put("c", 0.1, 0.1); // c: 10%
129    TransTable.put("d", 0.1, 0.1); // d: 10%
130    TransTable.put("e", 0.1, 0.1); // e: 10%
131    TransTable.put("f", 0.1, 0.1); // f: 10%
132    TransTable.put("g", 0.1, 0.1); // g: 10%
133    TransTable.put("h", 0.1, 0.1); // h: 10%
134    TransTable.put("i", 0.1, 0.1); // i: 10%
135    TransTable.put("j", 0.1, 0.1); // j: 10%
136    TransTable.put("k", 0.1, 0.1); // k: 10%
137    TransTable.put("l", 0.1, 0.1); // l: 10%
138    TransTable.put("m", 0.1, 0.1); // m: 10%
139    TransTable.put("n", 0.1, 0.1); // n: 10%
140    TransTable.put("o", 0.1, 0.1); // o: 10%
141    TransTable.put("p", 0.1, 0.1); // p: 10%
142    TransTable.put("q", 0.1, 0.1); // q: 10%
143    TransTable.put("r", 0.1, 0.1); // r: 10%
144    TransTable.put("s", 0.1, 0.1); // s: 10%
145    TransTable.put("t", 0.1, 0.1); // t: 10%
146    TransTable.put("u", 0.1, 0.1); // u: 10%
147    TransTable.put("v", 0.1, 0.1); // v: 10%
148    TransTable.put("w", 0.1, 0.1); // w: 10%
149    TransTable.put("x", 0.1, 0.1); // x: 10%
150    TransTable.put("y", 0.1, 0.1); // y: 10%
151    TransTable.put("z", 0.1, 0.1); // z: 10%
152    TransTable.put("A", 0.1, 0.1); // A: 10%
153    TransTable.put("B", 0.1, 0.1); // B: 10%
154    TransTable.put("C", 0.1, 0.1); // C: 10%
155    TransTable.put("D", 0.1, 0.1); // D: 10%
156    TransTable.put("E", 0.1, 0.1); // E: 10%
157    TransTable.put("F", 0.1, 0.1); // F: 10%
158    TransTable.put("G", 0.1, 0.1); // G: 10%
159    TransTable.put("H", 0.1, 0.1); // H: 10%
160    TransTable.put("I", 0.1, 0.1); // I: 10%
161    TransTable.put("J", 0.1, 0.1); // J: 10%
162    TransTable.put("K", 0.1, 0.1); // K: 10%
163    TransTable.put("L", 0.1, 0.1); // L: 10%
164    TransTable.put("M", 0.1, 0.1); // M: 10%
165    TransTable.put("N", 0.
```

Unicode への変換テーブル（抜粋）

[illegible]

変換プログラム（抜粋）