前　　言

われわれの21世紀COEプログラム「東アジア世界の人文情報學研究教育據點」では“漢字文化の全き繼承と發展”をスローガンに掲げている。漢字による文化資源を全面的に繼承するためには、漢字文獻の各種データベースをコンテンツの面で量的に充實させるばかりでなく、更に新しい技術の開發による利便性の向上を目指すことが求められるのは當然であろう。しかも今後の漢字文獻データベースは、少なくとも東アジアの漢字使用圏における廣範な利用者を前提にして設計されねばならない。そのためには技術開發の面で、中國との交流を推進し、相互理解に努めること、ひいては日中間において存在し得る幾つかの問題について、その解決はともかく、現狀を正確に認識しておくことが緊急の課題であると思われる。そこで漢字文獻のデータベース化において中國で中心的な役割を果たしている中國國家圖書館とのあいだで、日中共同シンポジウム「漢字文獻資料庫的新技術」を開催することで、この課題に緒口を見いだそうと考えた。

幸いにして國家圖書館の陳力副館長、及び嚴向東國際交流處長をはじめとする同館スタッフの熱心な協力があり、約半年に及ぶ準備の末、2005年1月22日（土）、嚴しい寒さのなか、國家圖書館にほど近い湖北大厦を會場として開催の運びとなった。漢字文獻データベースの技術的な問題に關して、日本側4名、中國側4名の報告が行われ、それらを中心に率直かつ活潑な意見交換を行うことが出來たことは、極めて有意義であった。本册子にはこれらの報告の改訂稿を、當日の報告順に從って掲載する。ただ中國側報告者の内、中國國家圖書館科研處處長孫一鋼（SUN Yigang）氏の「數字文獻處理的標準規範研究」については、同氏からの申し出によりここに收録することが出來なかったのは殘念である。

漢字文獻データベースの技術について日中の研究者のあいだで直接に意見交換を行う機會はこれまでも決して多くなかった。今後、この種の試みが繼續して行われ、東アジア的規模で漢字文化繼承の具體的問題が一層活潑に討議されることを願ってやまない。

2005年7月17日

高田時雄

目　次　Table of Contents

Text-Searchable Image and Its Applications

Koichi Yasuoka

Documentation and Information Center for Chinese Studies,
Institute for Research in Humanities, Kyoto University

1 Introduction

Since 1996 proposal of the Council for Science, the university libraries in Japan have progressed "The Digital Library Project". Nowadays the union catalogue database of the university libraries (NACSIS-CAT) is almost completely equipped, and we can easily find any books and magazines in the libraries through the database on the Internet. But we are still far and away from the goal of "The Digital Library Project", which is the digitalization of all the books and the magazines in the libraries. The university libraries have only made displays of images of the rare books without their digital texts, their digital tables of contents, or their digital indices. The digital libraries in Japan now are not "libraries" but something like "museums", since they don't give us the way to "read" the books digitally.

In this paper the author represents the concept of text-searchable images and its applications. The author shows two formats, Portable Document Format and Scalable Vector Graphics, to actualize text-searchable images, and also shows a JavaScript-based program "`ttext-kanbun`" to produce text-searchable images in these formats. The author contributes this paper toward the true progress of the digital "libraries".

2 Text-Searchable Images

In this section we examine two formats, Portable Document Format (PDF) and Scalable Vector Graphics (SVG), to actualize text-searchable images.

2.1 PDF for Text-Searchable Images

The author has studied long time about text-searchable images using PDF [2]. And Adobe adopted some results of the study into PDF-1.4 [3] as "transparent text". Now we have two ways to actualize text-searchable images using PDF. The one is to put a transparent text upon an image, and the other is to put an image upon a text written in white characters. The former way is only available with the browsers of PDF-1.4 and after, and the latter way PDF-1.2 and after. In this paper we use the latter way for backward compatibility.

PDF can represent both images and texts, but has some limitations on its format. PDF supports only two compression methods for color images, that are JPEG and ZIP. PDF supports several character-sets for CJK texts, Adobe-Japan1-6 [7] (including 14663 漢字 characters), Adobe-GB1-4 [1] (including 27629 汉字 characters), Adobe-CNS1-4 [5] (including 17625 漢字 characters), and Adobe-Korea1-2 [6] (including 4620 漢字 characters) under Japanese, mainland Chinese, Taiwanese, and Korean circumstances, respectively. We need "Japanese Language Pack" to read and search PDFs written in Adobe-Japan1-6 character-set, so as mainland Chinese, Taiwanese, and Korean. This means that these character-sets are incompatible with one another, and that PDFs for text-searchable images actually cannot get across the borderlines. In this paper we use JPEG for color images and Adobe-Japan1-6 character-set for texts to produce text-searchable images with PDF.

2.2 SVG for Text-Searchable Images

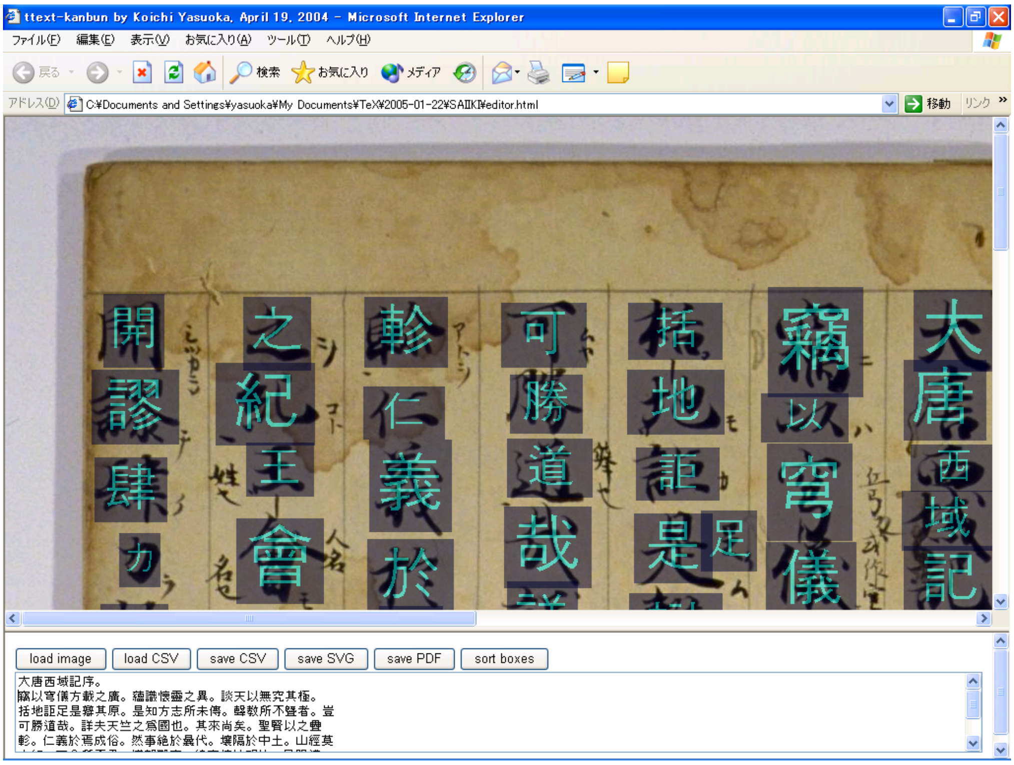
Tomohiko Morioka has studied about text-searchable images using SVG [4]. He actualized a text-searchable image to put an image upon a text. But in this paper we put a transparent text upon an image to actualize a text-searchable image using SVG.

SVG can include both images and texts, but the most contemporary viewer "Adobe SVG Viewer 3.0" has some limitations. SVG supports any kind of formats for color images, but the viewer suports only JPEG, PNG, and GIF. SVG supoorts any text-encodings but prefers UTF-8. In this paper we use JPEG for color images and UTF-8 for texts to produce text-searchable images with SVG.

3 Experiment and Result

The author wrote a JavaScript-based program "`ttext-kanbun`" to produce text-searchable images using PDF or SVG. "`ttext-kanbun`" runs on Internet Explorer 6 under Microsoft Windows XP.

We, members of COE21-project at Institute for Research in Humanities, Kyoto University, tried to make text-searchable images of 大唐西域記 (ex-橘寺-collection) with “`ttext-kanbun`” (Figure 1). We prepared 319 JPEG images for 大唐西域記, where each image has 2100×1950 pixels and total size of all images is 196807821 bytes, and its text written in UTF-8 consisting of 104725 characters (3138 different).

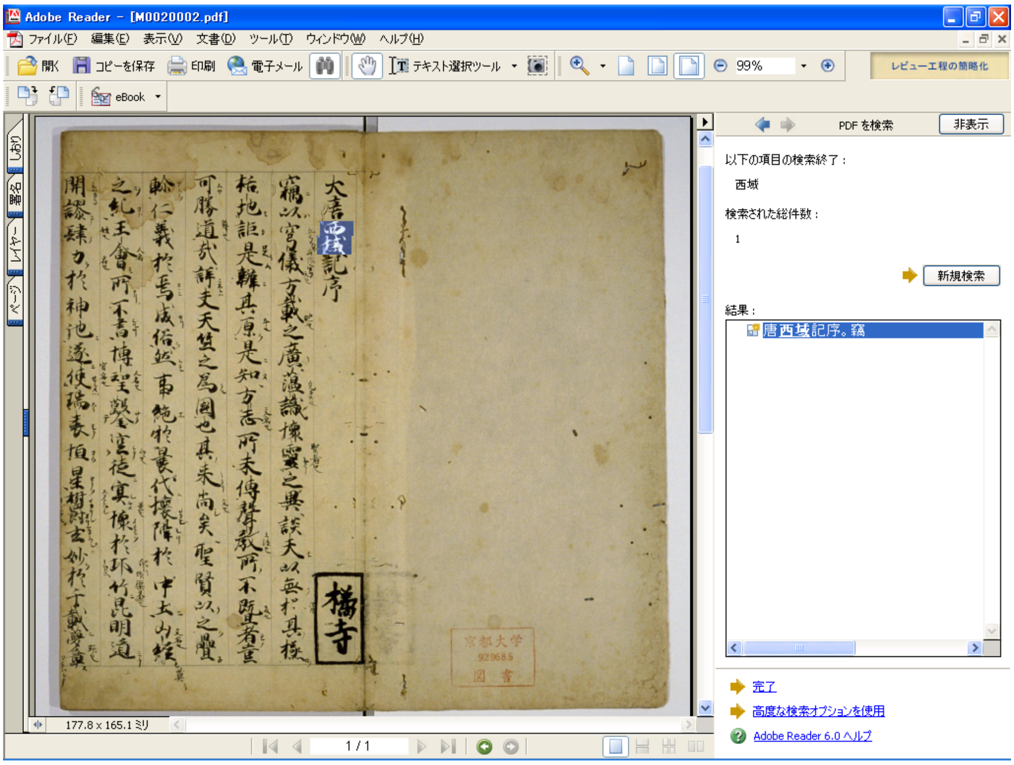

ttext-kanbun by Koichi Yasuoka, April 19, 2004 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り メディア
アドレス(D) C:¥Documents and Settings¥yasuoka¥My Documents¥TeX¥2005-01-22¥SAIIKI¥editor.html
移動 リンク
load image
load CSV
save CSV
save SVG
save PDF
sort boxes

大唐西域記序。
竊以穹儀方載之廣。蘊識懷靈之異。談天以無究其極。
括地詎足辯其原。是知方志所未傳。聲教所不暨者。豈
可勝道哉。詳夫天竺之爲國也。其來尚矣。聖賢以之疊
軫。仁義於焉成俗。然事絶於曩代。壤隔於中土。山經莫

Figure 1: Snapshot of “`ttext-kanbun`”

First we produced text-searchable images using PDF (Figure 2). The total size of 319 PDF files was 202662390 bytes, 2.97% increasing from original JPEG images. We couldn’t write 390 characters out of 104725 using PDF since they were not included in Adobe-Japan1-6. The 390 characters consisted of 51 different characters shown in Table 1. Then we combined the 319 PDF files into a multi-page PDF. The file-size of the combined PDF was 202440575 bytes, 2.86% increasing from original JPEG images.

Second we produced text-searchable images using SVG (Figure 3). The


Adobe Reader - [M0020002.pdf]
ファイル(F) 編集(E) 表示(V) 文書(D) ツール(T) ウィンドウ(W) ヘルプ(H)
開く コピーを保存 印刷 電子メール テキスト選択ツール 99% レビュー工程の簡略化
eBook
しおり 署名 レイヤー ページ
177.8 x 165.1 ミリ
1 / 1
PDF を検索 非表示
新規検索

以下の項目の検索終了：

西域

検索された総件数：

1

結果：

唐西域記序。竊

完了

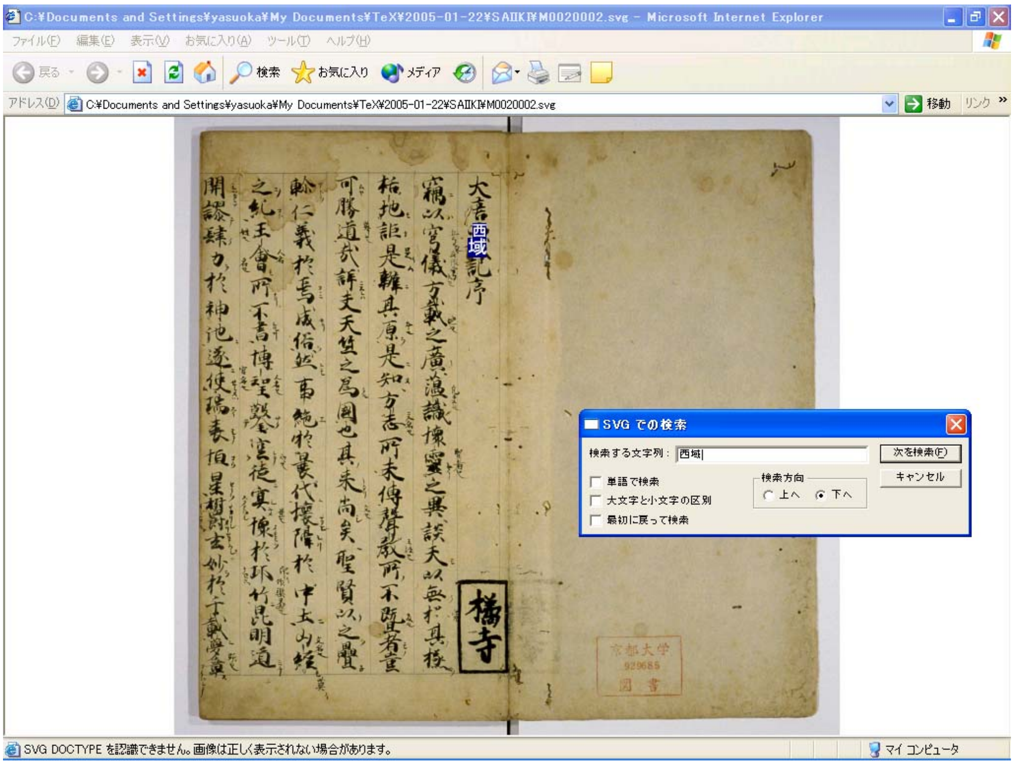
高度な検索オプションを使用

Adobe Reader 6.0 ヘルプ

Figure 2: Searching “西域” on “Adobe Reader 6.0”

偘 㔫 呾 壀 蓯 𥈠 𢈛 睹 噭 樬 櫰 𣨼 礔 礕 𥬝 𦺻
𦱳 𧂧 𧄸 𧶠 𡚱 隒 霫 顉 𪀁 鴐 黳 𢙛 𢺵 𣗥 㲉 㴔
𥢶 覔 𩣯 鷰 齏 𠡍 𠥼 𠮦 㦕 𢫘 𣀔 𤭛 𤎖 𦁡 卉 𦫵
𧘂 𩚅 䬉

Table 1: Characters not in Adobe-Japan1-6



C:¥Documents and Settings¥yasuoka¥My Documents¥TeX¥2005-01-22¥SAIKI¥M0020002.svg - Microsoft Internet Explorer

アドレス(D) C:¥Documents and Settings¥yasuoka¥My Documents¥TeX¥2005-01-22¥SAIKI¥M0020002.svg

大唐西域記序

竊以穹儀方載之廣蘊識懷靈之異談天以無究其極

括地詎是辯其原是知方志所未傳聲教所不暨者豈

可勝道哉詳夫天竺之爲國也其來尚矣聖賢以之疊

軫仁義於焉成俗然事絶於曩代壤隔於中土山經

之紀王會所不書博望鑿空徒寘懷於邛竹昆明道

閉謬肆力於神池遂使瑞表恒星鬱玄妙於千載夢

SVG での検索

検索する文字列： 西域

□ 単語で検索
□ 大文字と小文字の区別
□ 最初に戻って検索

検索方向 ○ 上へ ● 下へ

SVG DOCTYPE を認識できません。画像は正しく表示されない場合があります。

Figure 3: Searching “西域” on “Adobe SVG Viewer 3.0”

㔟	㔫	㖨	㦄	㩁	㪅	㲱	㶚	䋝	䒑	䒾	䘵	䭃	䭐

Table 2: Invisible characters on “Adobe SVG Viewer 3.0”

total size of SVG files and JPEG images was 203752662 bytes, 3.53% increasing from JPEG images only. All characters could be represented in SVG files, but 14 characters shown in Table 2 couldn't be displayed on "Adobe SVG Viewer 3.0", since the Viewer didn't support Unicode Plane-2 fonts.

4 Conclusion

In this paper the author has represented the concept of text-searchable images and its actualization using PDF or SVG. The author wrote a JavaScript-based program "`ttext-kanbun`" to produce such text-searchable images. As a result we have found that only 3% to 4% file-size increase is needed to add texts on JPEG images. The author now distributes "`ttext-kanbun`" at `http://coe21.zinbun.kyoto-u.ac.jp/~yasuoka/ftp/program/` and is pleased to help anyone to produce such text-searchable images.

References

[1] Adobe-GB1-4 Character Collection for CID-Keyed Fonts, Technical Note #5079, Adobe Systems (November 2000).

[2] Koichi Yasuoka and Tokio Takata: Digital Rubbings — Their Past and Future, 2001 Pacific Neighborhood Consortium Proceedings (January 2001), ECAI Rubbings Work Session.

[3] Adobe Systems Incorporated: PDF Reference third edition — Adobe Portable Document Format Version 1.4, Addison-Wesley (December 2001).

[4] 守岡知彦: ポスト文字コード時代の文書処理技術に関する展望, 全国文献·情報センター人文社会科学学術セミナーシリーズ, No.12 (November 2002), pp.59-70.

[5] Adobe-CNS1-4 Character Collection for CID-Keyed Fonts, Technical Note #5080, Adobe Systems (May 2003).

[6] Adobe-Korea1-2 Character Collection for CID-Keyed Fonts, Technical Note #5093, Adobe Systems (May 2003).

[7] Adobe-Japan1-6 Character Collection for CID-Keyed Fonts, Technical Note #5078, Adobe Systems (June 2004).

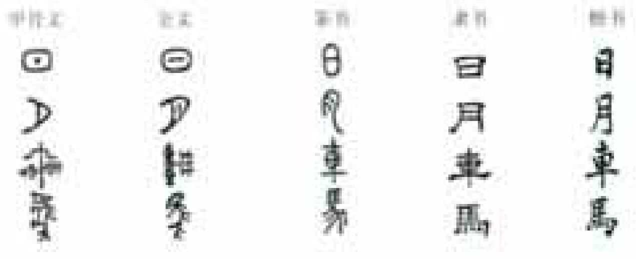
古籍数字图书馆的中文信息处理技术综述

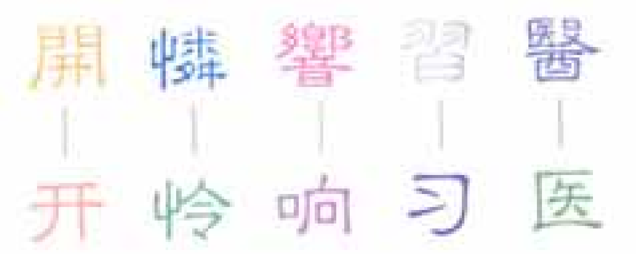
孙卫
中国国家图书馆

目 录

第一章　概述

中国现存最早的文字是甲骨文。距今已有 3500 年的甲骨文就是刻在龟甲或兽骨上的文字。主要是商代的文字。





上图是甲骨文到现在汉字的演变过程示意。

中国出土的最早的纸质绘画是甘肃玉门发现的，距今大约 1600 年。

我国最早的竹木简古写本是《仪礼》。1959 年考古学家在武威汉墓中发现了 504 根竹、木简。其中有 469 根，据初步考订，是西汉末年所抄写的《仪礼》。我国已发现的最早的比较完整的竹木简是战国时代的[1]。

我国现存最早的帛写书是《缯书》。《缯书》距今已有两千多年。l942 年 9 月在湖南长沙东郊子弹库的纸源冲的战国楚墓中出土。这是一件用毛笔墨书、彩绘在丝织品上的帛书，高约 30 厘米、长约 39 厘米。帛的中间写有长篇文字，分左右两部分。左方十三行，右方倒写

[1] 中国国家图书馆陈力馆长纠正

11 行，共计六百多字。所用字体为战国时代的古文，有的文字漫漶不清，多不可识。文字四周有植物、怪兽、三头戴角人像等十二龚。图像间注有说明文字。《缯书》于 1946 年被美国人柯克思诓骗掠夺到华盛顿，现藏耶鲁大学图书馆。

我国现存最早的纸写书是晋人手抄的《三国志》。手抄本《三国志》是陈寿撰成后不久抄写的。现有甲乙两种抄本。甲本于 l924 年在新疆鄯善县出土，是《吴书·虞翻传》、《吴臧张温传》的部分内容，共计 80 行、1090 余字，中有残缺。原本流入日本，国内有新印本流传；乙本于 1965 年 1 月在新疆吐鲁番县的英沙古城附近的一座佛塔遗址中发现，是《吴书。吴主权传》和《魏书·臧洪传》的残卷，共 40 行，计有 570 余字，中有残缺。甲、乙两种抄本均隶书体、行款恭正，但非一人抄写。乙本抄书年代早于甲本，但相距时间不会太长。

我国现存最早的木刻印本书是《陀罗尼经咒》。这是一张唐刻梵文经咒，1944 年 4 月出土于成都市东门外望江楼附近的唐墓。印本长 34 厘米、宽 3l 厘米，用唐代名茧纸印制，质地薄而透明，韧性强。印本中间小方栏内刻一菩萨像。栏外四周有数行梵文经咒。梵文外，又雕双栏，其中四角及每边刻有菩萨，像之间为佛教供品的图像。根据印本右边的题汉文可知，这张经咒是成都府成都县龙池坊卞家印卖咒文。据考证，《陀罗尼经咒》为唐代末期作品，不早于 757 年，印刷于公元 850 年以后的可能性大。此书现藏中国历史博物馆。

我国文献资料中提到的最早的雕版印刷书是唐朝的《女则》。根据明朝邵经邦《弘简录》一书的记载，唐太宗的皇后长孙氏编了一本

书，名叫《女则》。贞观十年，长孙皇后死去，宫中有人把此书送给唐太宗。唐太宗看到这本书讲的都是封建社会中妇女典型人物的故事，宣扬封建伦理道德，对巩固封建王朝的统治有好处，就下令用雕版印刷把它印了出来。

我国和世界现存最早的有日期的雕版印刷的书是《金刚经》。《金刚经》印成于唐咸通九年(公元868年)。它是一部长约1丈、6尺、高约1尺的卷子，由六张面积相等的印有经文的纸粘缀而成。卷首另有二幅扇画，画着释迎牟尼佛在祗树给孤独园的说法图。其余为《金刚经》全文，题有“咸通九年四月十五日王　为二亲敬造普施”一行。这卷举世闻名的《金刚经》，原藏甘肃敦煌千佛洞，1899年发现，1907年被英人斯坦因盗去，现藏伦敦不列颠博物馆。

西漢初年，政治穩定，思想文化十分活躍，對傳播工具的需求旺盛，紙作為新的書寫材料應運而生。許慎著的“說文解字”，成書於西元100年。

尽管中文的文字历史有3500年以上，计算机的历史也有100多年，而我国在计算机进行中文文字处理是在20世纪70年代中期的国家748工程开始的，目的是利用计算机和激光照排机替代铅字印刷进行出版。在20世纪80年代中期，电子工业部计算机总局开始组织个人计算机设计研究工作，在长城0525计算机中014/015板的设计中，首次实现了硬件的中文显示技术与西文显示技术的合一，完成了CCDOS中文汉化操作系统的研制等工作。在Windows 3.1操作系统

上实现中文是采用的外挂中文环境的方式，自从 Windows 3.2 操作系统以后，操作系统都采用了内核汉化技术，在 Windows 95、Windows 97、Windows 98、Windows ME 等都是采用不同语言的国家采用不同的内核处理语言的技术。Windows NT 4.0 以后，Windows 2000 和 Windows XP 都采用了底层采用 Unicode 编码进行文件的存放，在表现层可以支持各个国家不同语言标准的内核与表现层相结合的技术。

基本的中文处理技术是：1，中文的计算机输入技术。在键盘输入技术中有音码、形码、混合三种输入方式。鼠标输入中有部首、部首+笔画等输入方式。语音输入方式。光学字符识别输入方式。手写识别输入方式等。各种输入方法都是落后于字符标准的变化的。先有汉字编码，再对应各种输入方法的码表。2，中文汉字显示技术。主要是点阵显示方式、曲线显示方式、混合显示方式。在计算机屏幕上先采用的是点阵显示技术，现在发展到曲线显示或者混合显示技术。而早期的打印机是自己带各种字库来支持打印的，现在是靠曲线字来支持打印的。

对于中文文档和数据库的处理，需要更多的中文信息处理技术，主要是：1，中文检索技术。根据中文词表，对于文档或者数据库中的中文进行词切分，形成索引词表。再根据检索词在索引词表中寻找对应的词，根据这些词的位置找到相应的文档。由于中文很难由字的编码准确的判别检索词，所以中文检索技术是中文信息处理技术的难点之一。利用字符的完全匹配的计算机统配符技术很难满足中文检索的需要。2，中文检索结果集的排序。由于中文信息编码是逐步增加

和变化的，所以，到现在为止很难有一个统一的中文排序方法，拼音、笔画、部首等的排序方法都存在一定的缺陷。3，中文关联检索。在使用 Unicode 以后，如何对于简-繁关联进行检索就成为使得各种需要的人都能检索到满足自己需要的文档。

更难的是中文语义分析，进行中文概念的判断，中文文档的自动分类、标引、主题提取等是更难的中文信息处理技术。比较典型的是汉字联想输入[2]、汉字句子[3]输入并自动根据意思纠错。

对于国家图书馆，大量是现代书籍与古籍并存。现代的书籍用字范围大概在 3 万-3.5 万汉字之间[4]。而《康熙字典》[5]的正字范围在 4 万 7 千字左右，《汉语大字典》[6]的正字范围在 5 万 6 千字左右。台湾整理出来的正字与异体字[7]大约在 10 万 6 千字左右。日本整理出来的汉字[8]大约在 5 万字左右。韩国当年处理《高丽藏》[9]使用了 2 万个左右的汉字。即使是Unicode 4.0 / ISO 10646-2003 也只有 71000 汉字。国家图书馆最近实验的《数字方志项目》又整理出了 4 千多个在地方志中出现的汉字。对于古籍的数字化处理过程中，除了用字数量以外，大量的避讳字也是一个难以处理的问题，所以古籍用字范围是一个主要的难点。在中国国家标准中，GB13000 是GB2312/GBK/Big5 等字形的一个包容集，采用的是ISO10646 的编码规范，对于字形进行过

[2] 微软汉语拼音输入法 2003

[3] 拼音之星 2000

[4] 国家新闻出版总署 2002 年研究项目，对于 1949 年以后，新闻出版所使用的汉字范围进行统计，统计的结果在 3 万到 3 万 5 千汉字之间——北京中易公司（项目的技术支持单位）。

[5] 同文书局石印版

[6] 四川辞书出版社与湖北辞书出版社联合出版

[7] http://140.111.1.40/start.htm

[8]《大汉和辞典》诸桥辙次

[9] 《高丽藏》电脑工程考察记 乌.阿浦 著 刘建 译

相应的规范。在这 20902 个汉字（ISO10646 基本集）、扩充A 6582 个汉字，总计 27494 个汉字是可以拼音输入[10]的。但是，对于扩充B的 4 万 3 千多个字是无法用拼音输入的。所以的汉字的输入方法也是古籍数字化的难点之一。光学识别（OCR）技术，在中国进行的大量技术研究，目前比较理想的识别率是对于铅字印刷的文字作品，对于比较规范的抄本或者刻板识别率很难做到 99.997%，识别的字大约在 1 万 5 千个汉字。由于中国 5 千年的历史，遗留的文字作品很多，有些作品并没有一个规律可以去发现和学习，利用技术中的学习方法找到其中规律，提高准确率是很难做到的。对于古籍的利用，如果不能达到一定的准确率，全文化就失去了很多研究的价值。古籍的检索与现代文的检索技术也有很大的差异，现代文一般强调词和复合词的检索技术，所以词切分就非常重要。而古籍在大部分时间是字或字的组合检索，所以字的切分与关联就是最重要的。由于古籍的用字与计算机规范的字是有差异的，所以全文与古籍全文图像的对应位置的对应就非常重要。这些都是现代文档和古籍在中文信息处理技术上的差异。

对于一个古籍数字图书馆的基本要求如下:

1， 一个 B/S 结构的古籍数字资源的发布、服务、研究系统。古籍数字化的目的不是为了古籍的保存，这点是非常明确的，古籍数字化最主要的是把人类文化遗产最大限度的揭示出来，并能够在揭示的基础上做研究。全世界对于中文的研究尤其对于古籍的研究，是对中国历史的认识的加深，所以为了方便这些学

[10] 香港华通《拼音大师》

者把注意力放在研究古籍上，而不是计算机操作上，古籍数字资源的发布、服务、研究需要建立在B/S结构上，只要会上网就能进入研究。

2， 在这个系统上，客户端要解决轻量级的曲线字、多种中文输入法、标准字库和自定义字库等基本中文信息处理技术。

3， 在这个系统上，服务器端要解决中文检索、全文与图像的文字位置对应、中文关联检索、打印输出控制、用户管理与授权等中文信息处理技术和安全控制技术。

4， 内容上有古籍的图像、全文、知识库等部分构成。

第二章　基本中文信息处理技术

一、编码体系关系

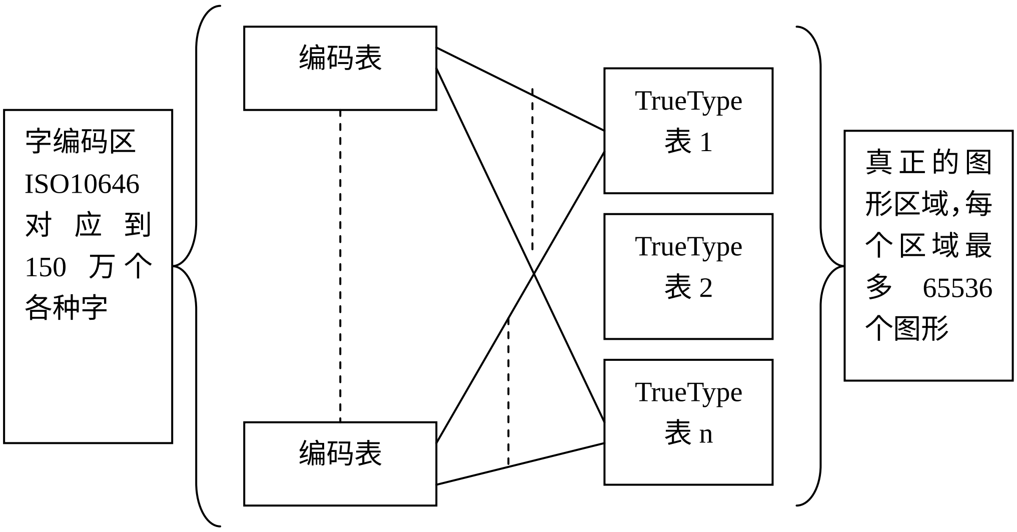

字编码区 ISO10646 对应到 150 万个各种字
编码表
编码表
TrueType 表 1
TrueType 表 2
TrueType 表 n
真正的图形区域，每个区域最多 65536 个图形

图 1、字形与编码的关系示意

在图 1 中，示意出了计算机的曲线字库与编码表之间的关系。由于 ISO10646 是变字节编码技术，所以，在这个编码体系中可以容纳 150 多万个各种字。所以，ISO10646 标准是编码和形同时具备的字符集，在这个字符集中可以同时容纳与显示世界上所有的计算机上的文字。在古籍中要采用 ISO10646 的考虑，一个是字的数量大，一个是全世界研究文字的人可以交换各种文字的数据。

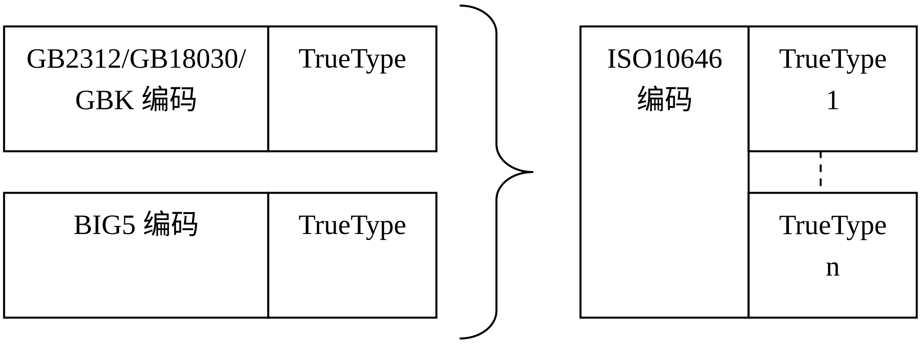


图 2、国家标准和台湾规范进入国际标准

在图 2 中，我们可以清楚地看到不同编码之间，完全不相同的部分是编码的码表，而曲线字现在是一样的。

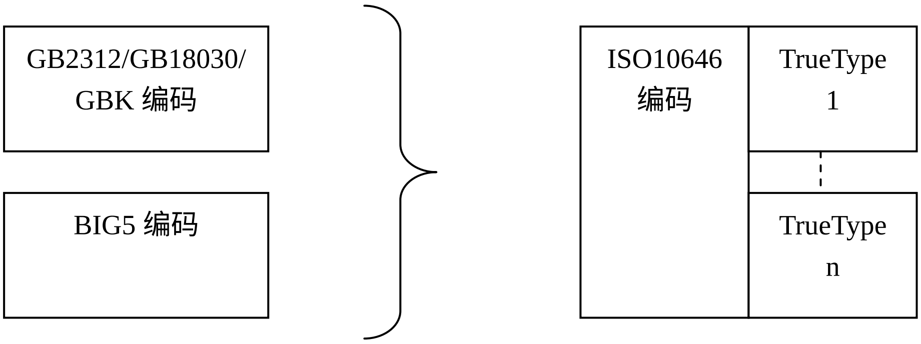


图 3、编码转换机制

在图 3 中，我们只需要采用一种曲线图形，而把不同的编码转换到ISO10646 编码就可以实现内核采用一种统一编码[11]，表现层采用不同的编码格式的目的。

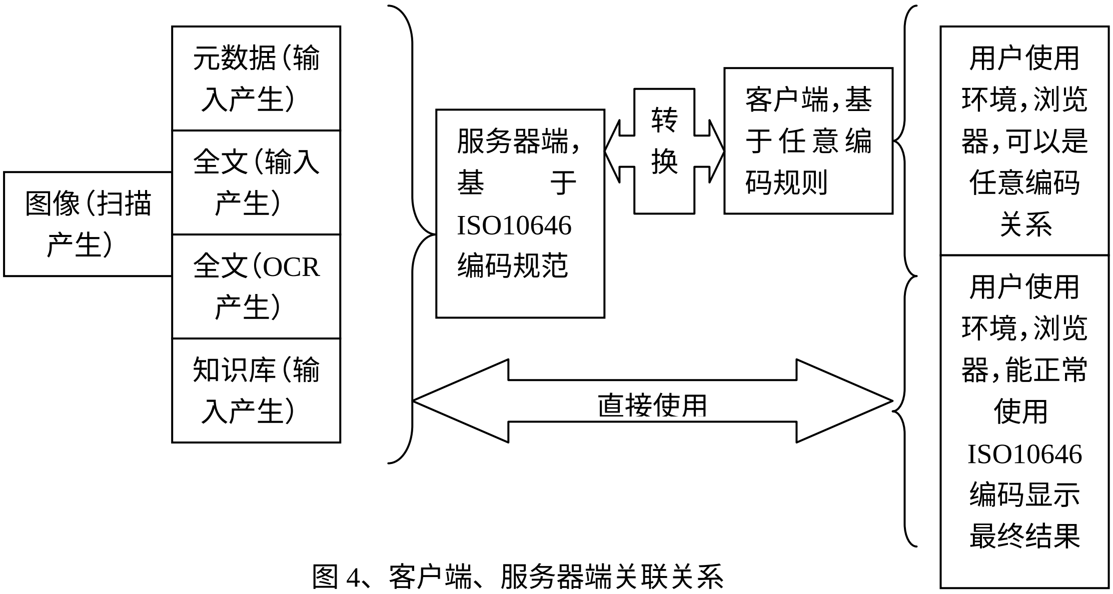

图像（扫描产生）
元数据（输入产生）
全文（输入产生）
全文（OCR 产生）
知识库（输入产生）
服务器端，基 于 ISO10646 编码规范
转换
客户端，基于任意编码规则
直接使用
用户使用环境，浏览器，可以是任意编码关系
用户使用环境，浏览器，能正常使用 ISO10646 编码显示最终结果

图 4、客户端、服务器端关联关系

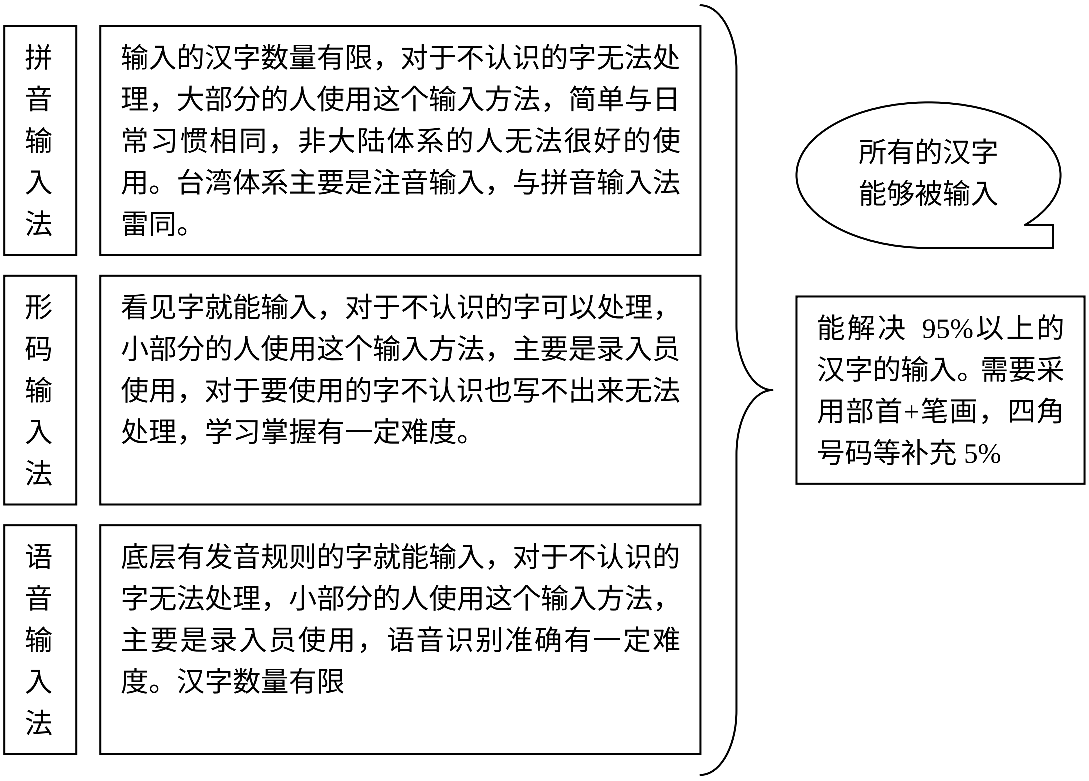
图 4 简要解释了 B/S 结构中编码转换的位置关系。对于世界各国都要研究古籍，就需要在底核采用 ISO 10646，在表现层可以灵活。

TrueType 曲线字库过大，不宜在 B/S 结构下传输字库与各种操

[11] 微软公司的Windows 2000 和Windows XP等就是采用这样一种方式方法

作系统支持下正常使用这个曲线字库。

二、输入法的关系



拼音输入法

输入的汉字数量有限，对于不认识的字无法处理，大部分的人使用这个输入方法，简单与日常习惯相同，非大陆体系的人无法很好的使用。台湾体系主要是注音输入，与拼音输入法雷同。

形码输入法

看见字就能输入，对于不认识的字可以处理，小部分的人使用这个输入方法，主要是录入员使用，对于要使用的字不认识也写不出来无法处理，学习掌握有一定难度。

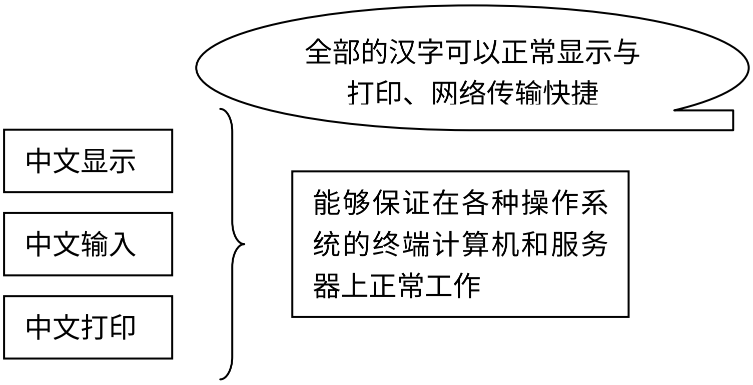
语音输入法

底层有发音规则的字就能输入，对于不认识的字无法处理，小部分的人使用这个输入方法，主要是录入员使用，语音识别准确有一定难度。汉字数量有限

所有的汉字能够被输入

能解决 95%以上的汉字的输入。需要采用部首+笔画，四角号码等补充 5%

图 5、主流输入方法的比较



全部的汉字可以正常显示与打印、网络传输快捷

中文显示

中文输入

中文打印

能够保证在各种操作系统的终端计算机和服务器上正常工作

图 6，中文支撑环境在各种操作系统的正常工作

图 5 表明了要使用中文的第一个重要的步奏就是要把汉字能够在各种计算机操作系统环境下输入进入浏览器并能正常的现实与打印

中文。图 6 是对于计算机支撑中文的最低要求，如果不能满足这个要求，也就无法做到在 B/S 结构下使用古籍数字图书馆系统。

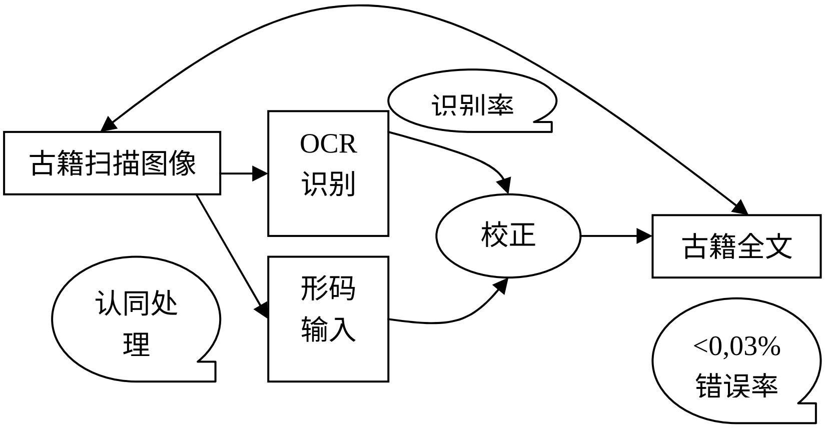


图 7、工业化输入的方式方法

图 7 是古籍中文全文化加工的主要的方式与方法。在这个过程中，困难的是保证全文化的错误率小于 0.03%。OCR的识别范围在 1 万 5 千字，处理近代的铅字版、稳定的刻版、抄版有一定的工业化的加工全文的能力[12]。人工输入处理时代范围大，刻版、抄版不一致的有一定的工业化加工能力，主要的错误来自于人为对于字的拆解和抄版的不清楚[13]，可以在 7 万 1 千汉字标准中进行录入处理。

还有一种输入方法就是手写识别，如果这种输入方法能够与 7 万 1 千汉字紧密结合，并保证识别率，将是一种受欢迎的输入方式。其他的传统输入方法搬进计算机，象不受输入法、笔画输入法、四角号码输入法等利用鼠标保证 7 万 1 千个不常用的字的输入也是非常有价值的工作。

[12] 香港迪志公司与中国书同文公司合作加工的《四库全书》
[13] 北京中易公司为国家图书馆进行的数字方志项目的实验

第三章　中文检索技术及其相关技术

一、基本检索技术

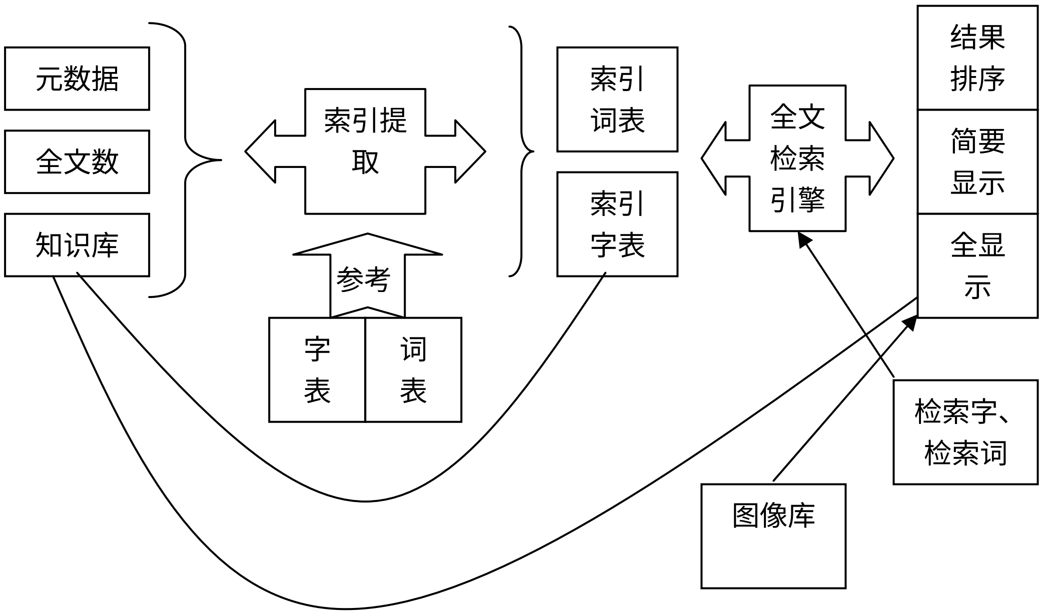


图 8、古籍全文检索系统结构关系示意

图 8 是一个全文检索系统结果的过程关系示意图，这个检索与现代文的检索有些不同，主要是字表和全文与图像的对应。在古汉语中，单词检索是对于古籍研究中的一个重要的检索方式。全文中检索字或者词的加亮与图像上对应的字或词进行加亮是另一个重要的用途，由于计算机用字是规范的标准，而古籍上的很多字是非规范的，所以必须方便古籍的研究者能够利用计算机进行检索，也可以根据检索到的结果对应原始的扫描图像进行比对研究。

二、关联的重要性

1，字关联

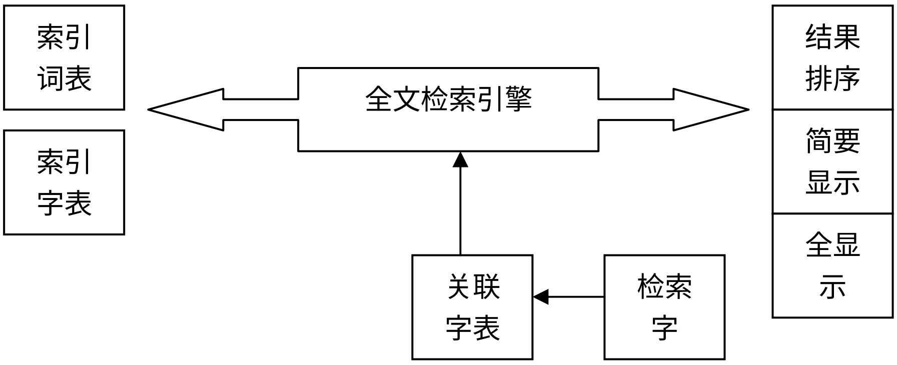


图 9、字关联位置关系示意

图 9 是一个字关联位置的关系示意。因为在 ISO10646 中，有大量的异形异体字，字关联主要是指简——繁关联，异形异体——同义关联。当研究者发一个简体字进入检索系统，先通过关联字表找到对应的繁体字，把简体、繁体字一起发到检索引擎中去，在索引字表中找到对应的位置，一并作为结果返回。这样的字关联可以极大的方便研究者找到自己知道或者自己不熟悉的字和同义异形异体字。作为古籍数字图书馆来说，这个字关联是非常重要的。

2，知识关联

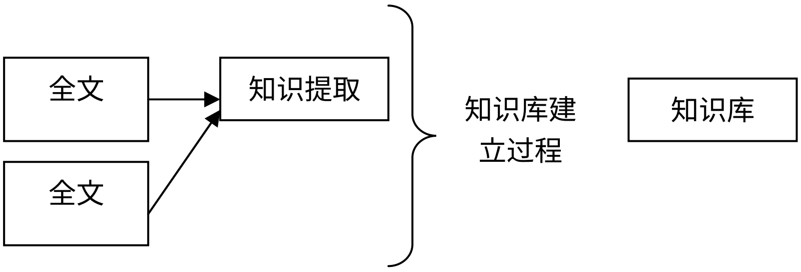


图 10、知识库建立示意

图 10 是知识库建立过程的示意。在一个全文中提出或者多个全文中提出的知识点，知识点的集合称为知识库。提取知识库是对于古籍

研究中重要的参考工作之一。

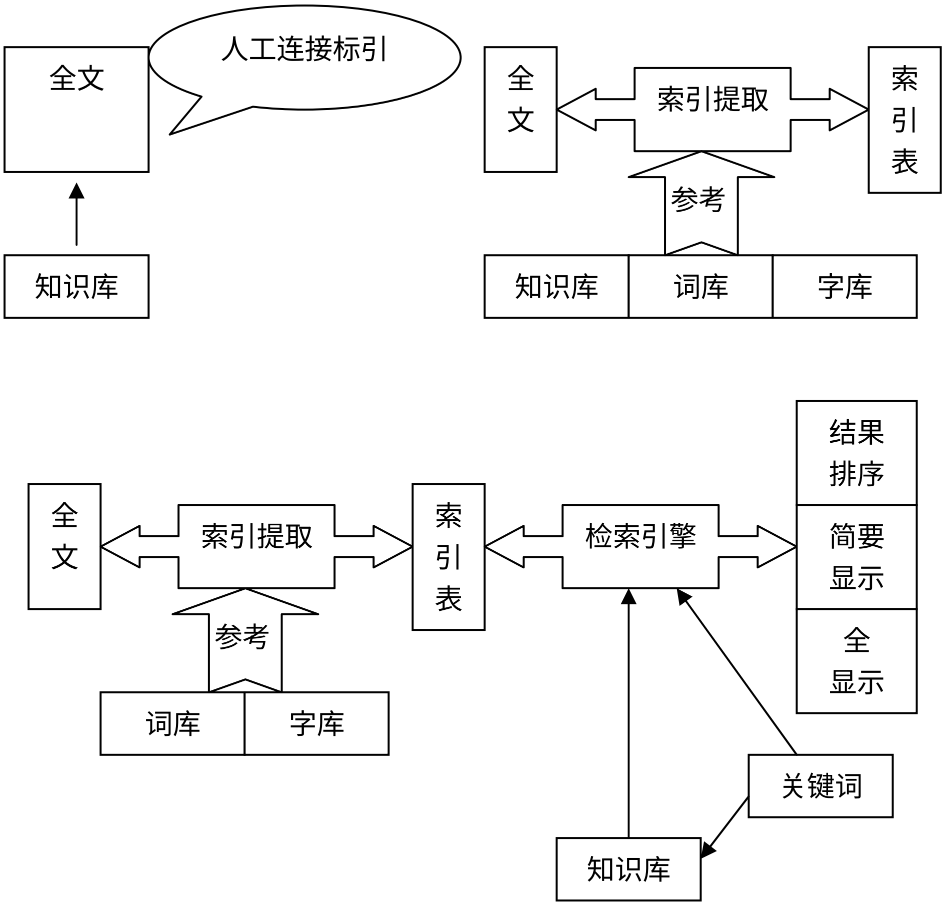


图 11、知识库多种用法示意

图 11 中示意了多种知识库的使用方法，一种是在全文中对于知识点进行标引、挂接。然后在正常的使用全文检索。这个方法好处是标引清楚、链接准确，但是带来一个问题，就是知识点随着时间的变化而增多时，要不断的对全文进行标引、链接。另一种是把知识库作为进行索引提取的依据，对于全文进行字、词、知识库参考索引表提取。检索者只要发检索词在某个知识点上，就会自动关联链接。缺点是知识库变化，就要重新抽取索引，当数据很大时，这个索引时间也是很长很长的。还有一种方法，就是对于检索关键词进行知识关联，再利用检索引擎中的词、字组合技术产生知识关联，这个方法的好处就是

对于全文、索引的变化小，而是需要研究一个关联分析和词、字组合技术来实现。

3、其他关联

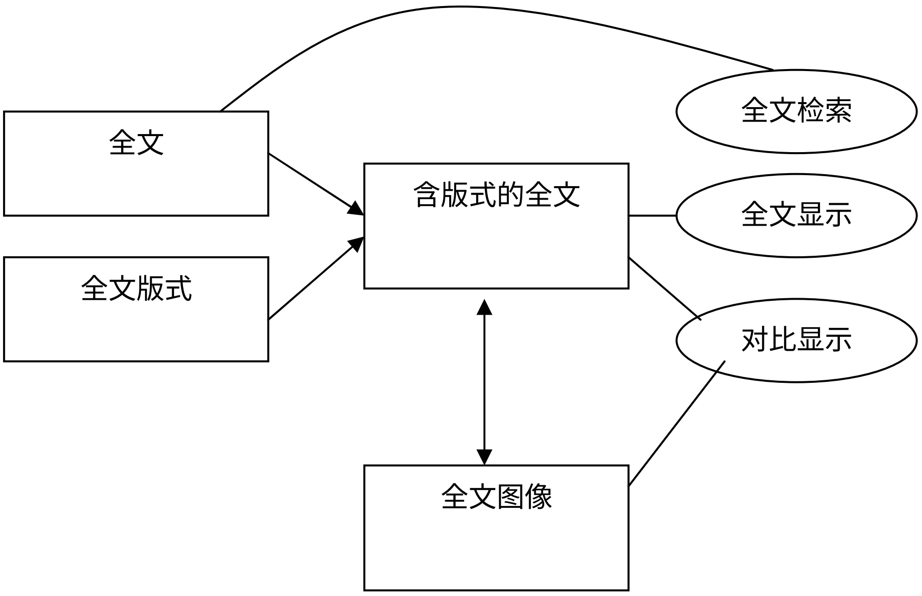


图 12、其他关联示意

图 12 是一个多关联的显示，OCR 或者全文输入后的全文，一般是参加检索用，现在增加一个版式关系以后，全文显示更象原来古籍的版型了。检索的词或者字在全文中加亮是一般能做到的，现在要把这个对应的加亮对应到全文图像和全文版式文件中，更好的方便使用者进行对比和比较。

第四章　古籍处理中的技术瓶颈

一、古籍全文的 XML 格式文档的结构化

对于古籍数字化加工后形成的图像、全文、版式、位置关系、知识点挂接等，需要一个合理的 XML 文档结构进行组织。这个结构化的文档作为数据交换的基础。

在这个文档中，至少应该包括以下的几个部分：

（1）头文件区里应该能指明字符集规范（编码规范、形体规范）

（2）版式描述区中，应该对于每个页的版式进行描述（坐标参考点、版式的分栏、分栏变量（列））

（3）内容描述区中，可以把元数据标明，然后把内容结构标明，对应的列变量与版式描述区中对应。

（4）在一个目录下，至少有一个总的 XML 结构，描述在这个目录下的各个 XML 文档的关系。

（5）应该标明对应的图像文件的标识关系。

二、曲线字的处理能力

在 B/S 结构中，主页中应该有一个 JAVA 的环境变量检查程序，探明本地计算机是否有支撑 ISO10646-2003 的标准字库，如果本地没有时，应该替换 XML 结构中的字库标准与规范为轻量的曲线字。这个轻量曲线字应该是曲线字标准，但是总的字节数很小，有利于网络下载，可以支持 B 上的缩放和打印等功能。不能在 XML 原始结构中

去调整字符的选择，所以增加一个判断和替换环境变量的方法来满足内容显示的需要。

三、中文输入能力的处理能力

在 B/S 结构中，主页中应该有一个 JAVA 的环境变量检查程序，探明本地计算机是否有支撑 ISO10646-2003 的标准字库的输入方法，如果本地没有大字符集输入方法时，应该在服务器中下载一个中文环境外挂包，有各种输入法，必须要有鼠标按照部首、笔画、四角号码等一般的输入方法。由于计算机环境是很多的，只有通过判断来增加中文环境支撑包。

四、字的处理能力

根据国家图书馆数字方志试验项目看，ISO10646 2003 标准中的 7 万 1 千多个汉字可以满足 95%以上的用字需求，由于中国有 3500 年文字的历史，古籍文献越古老，需要的汉字的数量将会增加。在数字方志中，增加了 4 千多个汉字。在国家图书馆另一个项目中，对于西夏文处理，专门制作了 6 千多个西夏文用字。而不断增加的字，需要标准化，这个是古籍全文化处理的一个难点。

还有一个问题，就是古籍中的避讳字的处理[14]，这个部分是在正字的基础上进行限制，如何使得古籍全文化中保持避讳字这个特别的历史事实也是很重要的。

由于中国文字历史有 3500 年历史，很多是手抄本、刻版的古籍、

[14] 《开放古籍平台的意义和实作》叶健欣

手稿，所以，因人而异的文字变化随处可见，既要保留历史事实，又要减少造字数量，所以辨别这些问题也是比较困难的。

五、古籍研究平台的使用推广

由于古籍处理的工作量很大，难度很大，而使用的人相对较少。为了解决这个使用人少的问题，开专门的 B/S 结构古籍研究平台有很大的价值，就是维系一套资源齐全的系统，而不需要每个有古籍研究者的地方都维护一个系统，这样可以节省整体的系统的维护成本。1999 年在美国的访问中，专门在美国国会图书馆东亚部、伯克利大学东亚图书馆、斯坦福大学东亚图书馆、哈佛大学燕京图书馆等，对于中文古籍的研究这些大学和单位都有相应的研究学者，但是这些研究者与中国和台湾、香港、新加坡等中文古籍的研究者在使用与操作上完全不同。所以，建立统一的古籍研究平台的难点是推广使用问题。

（1）美国的学者是用英文检索所需的资料，再使用中文的资料的检索、借阅资料的方式。新加坡学者中有一部分与美国学者一样，一部分与台湾、中国、香港等地学者的使用一样。台湾和香港主要是使用 BIG5 规范和注音、仓颉等输入方式检索中文。在中国有使用 GB 标准，也有使用 ISO10646 标准和音码、形码组合的输入方式检索中文。如何能够不改变各个学者原有的研究习惯来使用古籍研究平台是一个值得研究的问题。

（2）原始文献有一大部分是集中的，所以加工比较容易，但是，

由于历史原因，不少文献散落在世界各地，如何协调一致进行加工处理，再把检索用资源集中放置在古籍研究平台中，把图像分布在各个原始藏品地。这样的合作是建设好这个古籍研究平台的内容的关键。合作建设了就可以在研究中有优先权使用。

（3）多语言的一个学者讨论环境也是古籍研究平台所需要。数字古籍的主要价值是利用计算机技术来帮助学者进行古籍的研究，至少在古籍内容查找和做多古籍比较中，计算机的检索技术是一个重要的工具。计算机的文字的可编辑性与图像的可裁减形式研究者撰写文章的选择方法之一。多国学者对于某个共同的问题在研究平台上的讨论，也是对于全世界的汉学研究者提供一个对话的基础。而多语言支持能力是这个平台的关键。

六、古籍研究平台的国际合作

（1）内容制作上的合作。这个是古籍研究平台国际合作中最有价值的地方。具体的合作主要是：资源制作标准与规范的协调一致；元数据标准规范的协调一致。

（2）技术规范的合作。主要表现在输入技术、各个国家的编码技术到 ISO 10646 的转换对照技术、组字技术、检索技术等。

（3）人员上的交流与研究者之间的沟通，主要对于标准字、避讳字、组字、异型异体字的认同与意义研究的交流

第五章　国家图书馆的古籍研究平台的考虑

一、古籍内容数字化

（1）拓片、敦煌文献等数字化实验

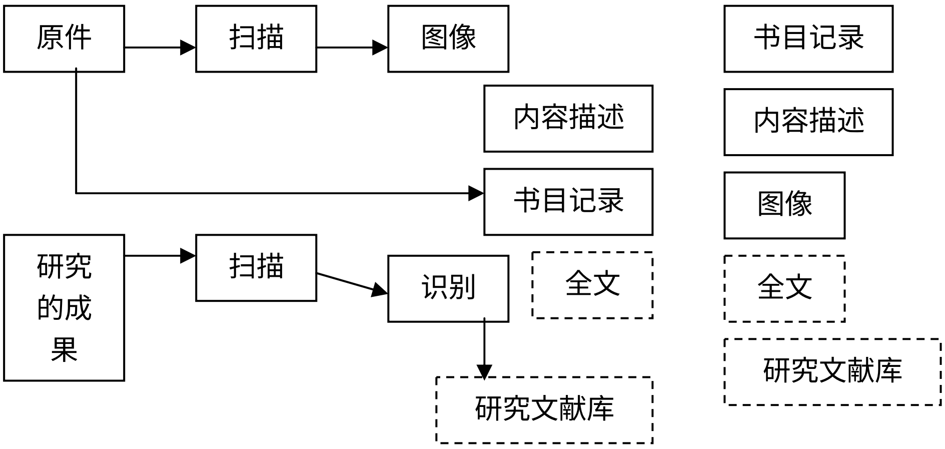


图 13、拓片数字化示意

（2）方志的数字化实验

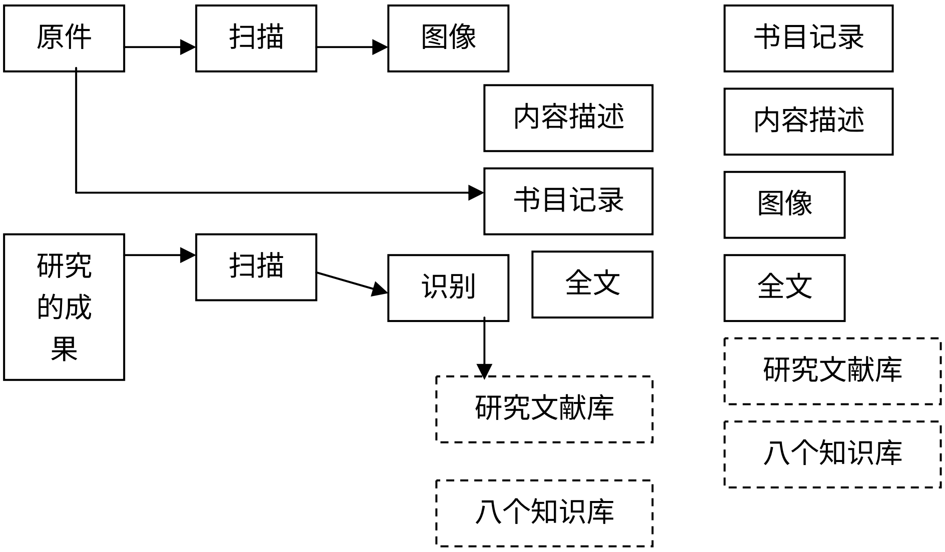
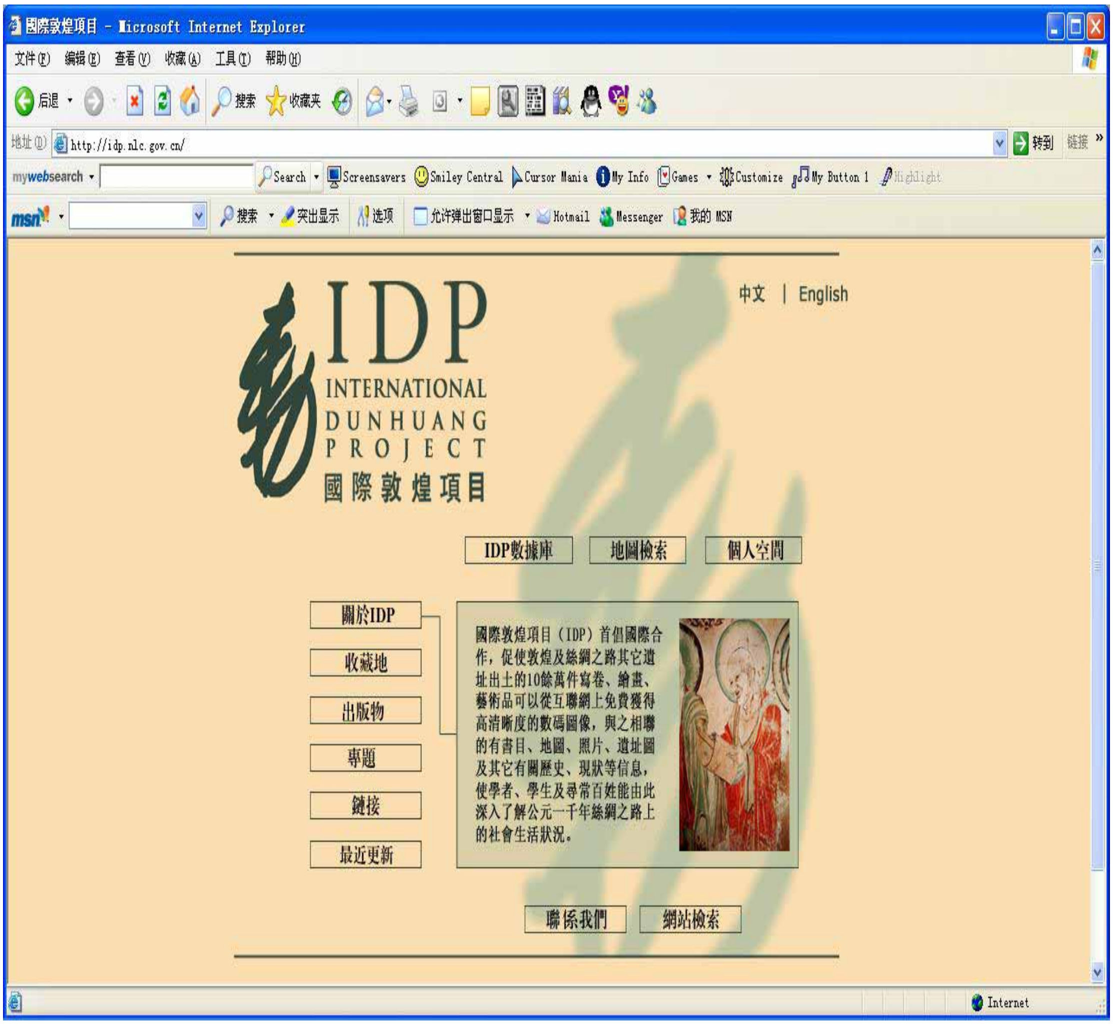


图 14、方志数字化示意

这两个部分的部分数字资源可以在国家图书馆的网络上看到。数字方志部分将逐步开放使用。

二、古籍研究平台实验

（1）http://idp.nlc.gov.cn/

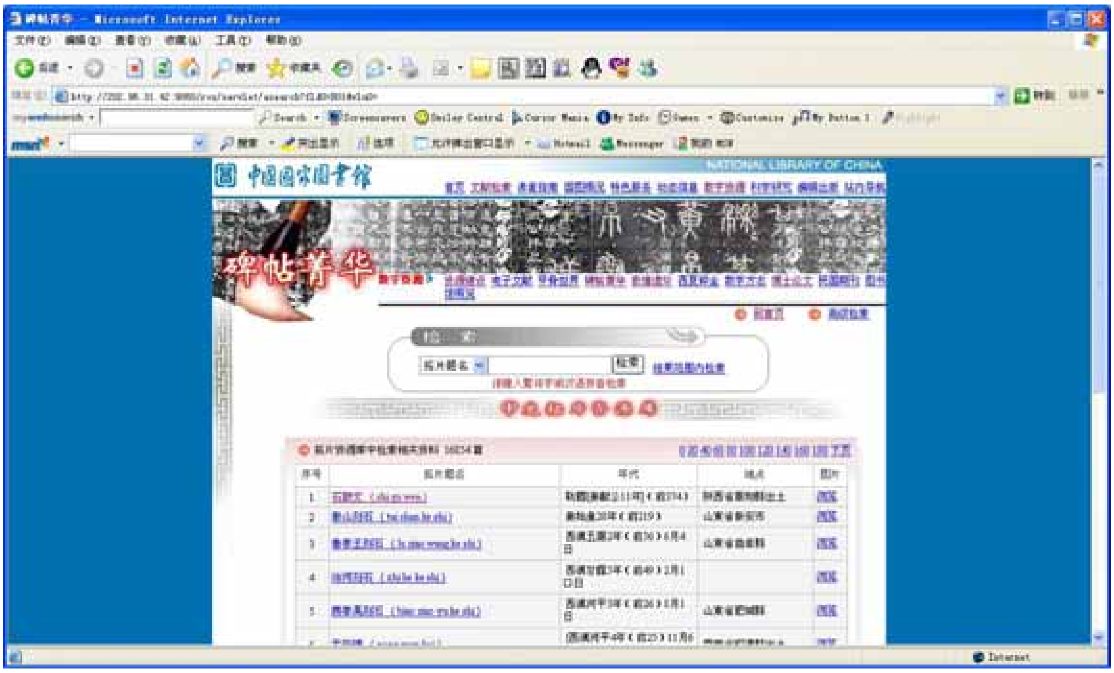


國際敦煌項目（IDP）首倡國際合作，促使敦煌及絲綢之路其它遺址出土的10餘萬件寫卷、繪畫、藝術品可以從互聯網上免費獲得高清晰度的數碼圖像，與之相聯的有書目、地圖、照片、遺址圖及其它有關歷史、現狀等信息，使學者、學生及尋常百姓能由此深入了解公元一千年絲綢之路上的社會生活狀況。

在这个项目中，检索采用的是多条件组合的排列式检索。就是在多个条件下，所有的结果排列出来，由使用者来选择。不是我们常用的检索方式。

检索到的结果基本是用图像挂接方式显示的。但是，链接的图像有一部分在中国国内，一部分在英国。这个是一个典型的对象分布的古籍研究平台，而且是国际合作的结果。目前，全世界的敦煌研究者都可以免费使用这个古籍研究平台的所有的数字资源。

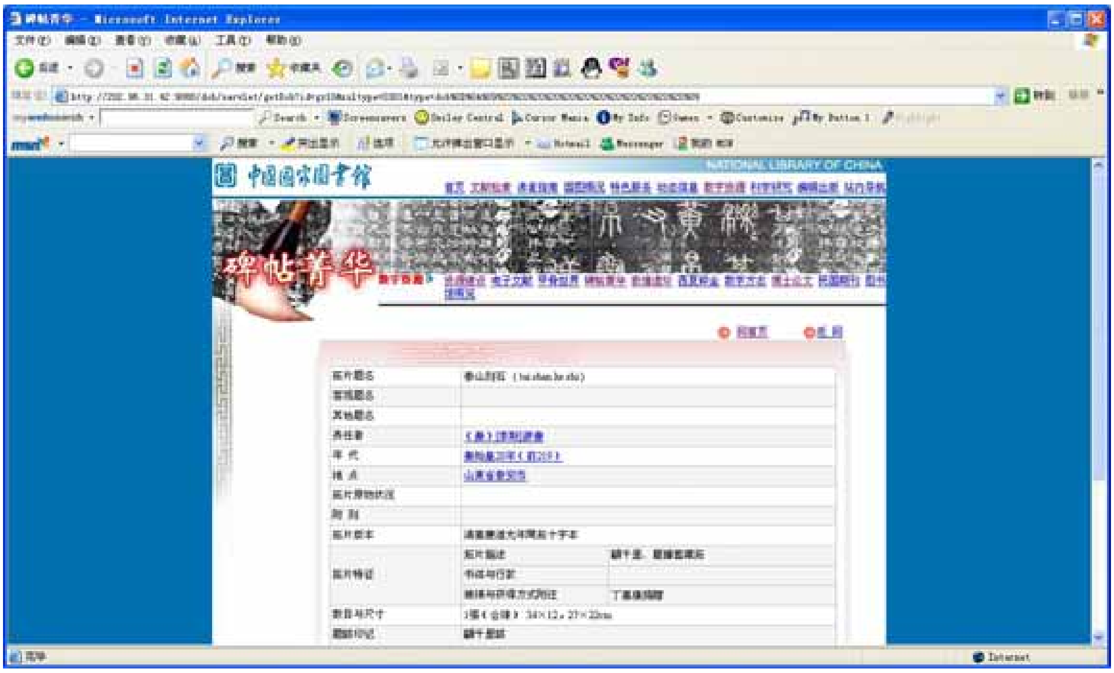
（2）http://202.96.31.42:9080/ros/index.htm



碑帖菁华 - Microsoft Internet Explorer
中国国家图书馆
NATIONAL LIBRARY OF CHINA
碑帖菁华
检 索
拓片题名
检索

序号	拓片题名	年代	地点	图片
1	石鼓文（shi gu wen）	[illegible]	陕西省[illegible]	图片
2	泰山刻石（tai shan ke shi）	秦始皇28年（前219）	山东省泰安市	图片
3	鲁孝王刻石（lu xiao wang ke shi）	西汉五凤2年（前56）6月4日	山东省曲阜县	图片
4	[illegible]刻石（[illegible] ke shi）	西汉甘露3年（前49）2月[illegible]日		图片
5	麃孝禹刻石（biao xiao yu ke shi）	西汉河平3年（前26）8月[illegible]日	山东省肥城县	图片

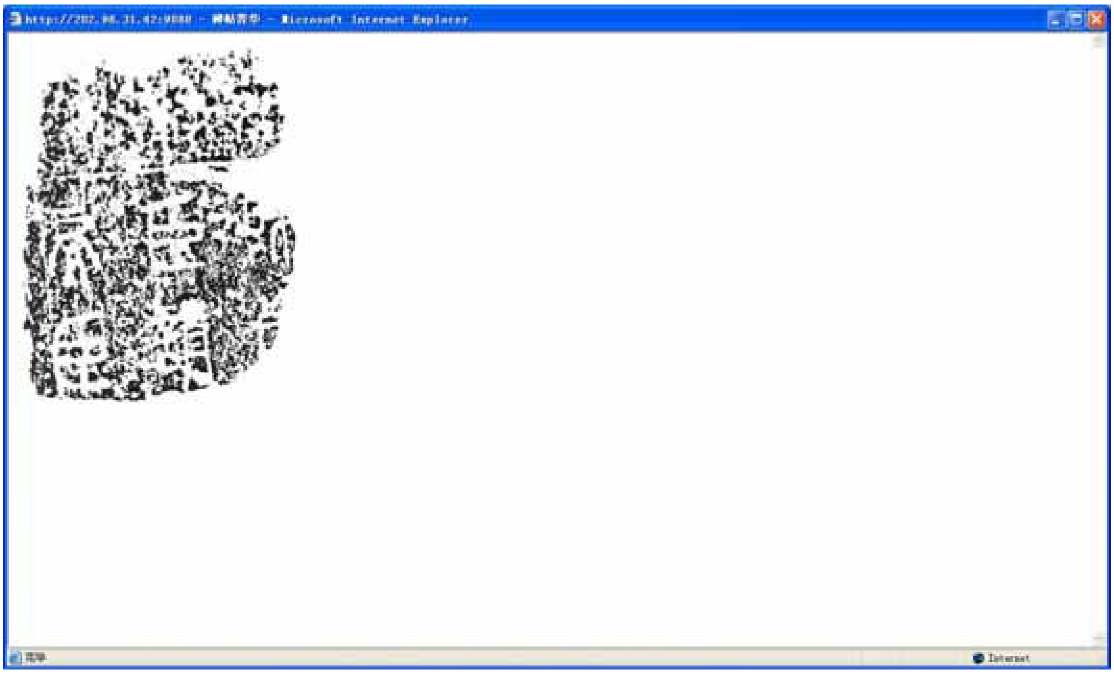
检索后，可以看内容描述，元数据



碑帖菁华 - Microsoft Internet Explorer
中国国家图书馆
NATIONAL LIBRARY OF CHINA
碑帖菁华

拓片题名	泰山刻石（tai shan ke shi）	
馆藏题名		
其他题名		
责任者	（秦）李斯[illegible]	
年 代	秦始皇28年（前219）	
地 点	山东省泰安市	
拓片原物状况		
附 注		
拓片版本	[illegible]十字本	
拓片特征	拓片描述	[illegible]
	书体与行款	
	装裱与拼接方式附注	[illegible]
数目与尺寸	1张（合装） 34×12，27×22cm	
题跋印记	[illegible]	

也可以看拓片


http://202.96.31.42:9080 - 碑帖菁华 - Microsoft Internet Explorer
完毕
Internet

3、http://202.96.31.42:9080/wenxian/

西夏碎金也是国家图书馆古籍研究平台的一个实验，基本的与拓片相同。

4、数字方志

后村居士集全文数据格式说明

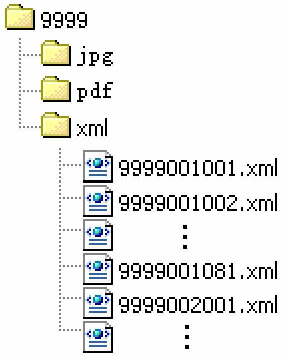
（1）数据组织

数据的命名方法：遵照中易古籍数字化加工系统的统一命名规则。数据以 ” 页 ” 为单位，数据的文件名格式为 10+3，即 10 位文件名加 3 位格式扩展名—— ” bbbbvvvppp.xxx ”。” bbbb ” 代表书目编号 BookID ；” vvv ” 代表卷编号 ；” ppp ” 代表卷内页号 ；” xxx ” 是扩展名。

提交的数据格式共有 3 种：*.jpg（原文图像数据）、*.xml（含格式描述信息的全文数据）、*.pdf（版式还原的电子书）。

在中易的古籍数字化库中，后村居士集的 BookID 为 9999，共分为 52 卷。3 种格式的数据分置于 3 个目录中，以下不再分目录。

数据组织如下所示。



```
9999
├ jpg
├ pdf
└ xml
    ├ 9999001001.xml
    ├ 9999001002.xml
    ├     ⋮
    ├ 9999001081.xml
    ├ 9999002001.xml
    └     ⋮
```

（2）全文数据（*.xml）格式

```
<?xml version="1.0" encoding="utf-16" standalone="yes"?>
<page pageid="9999001040" bookid="" relate="9999001040">
  <staffidx>
  </staffidx>
  <workflow>71</workflow>
  <info>
    <format count="6">
      <modulus id="imagewidth">1103</modulus>
      <modulus id="imageheight">1856</modulus>
      <modulus id="landscape">0</modulus>
      <modulus id="l2r">0</modulus>
      <modulus id="type">0</modulus>
      <modulus id="case">000</modulus>
    </format>
    <setting count="23">
      <modulus id="1" />
      <modulus id="2" />
      <modulus id="3" />
      <modulus id="4" />
      <modulus id="5" />
      <modulus id="6" />
      <modulus id="7" />
      <modulus id="8" />
      <modulus id="9" />
      <modulus id="10" />
      <modulus id="11" />
      <modulus id="12" />
      <modulus id="13" />
      <modulus id="14" />
      <modulus id="15" />
      <modulus id="16" />
      <modulus id="17" />
      <modulus id="18" />
      <modulus id="19" />
      <modulus id="20" />
      <modulus id="21" />
      <modulus id="22" />
      <modulus id="23" />
    </setting>
  </info>
  <data>
    <text count="10">
      <line id="1">
        <position>1070,190,966,1766</position>
        <content>    眞隱寺            道傍松</content>
      </line>
      <line id="2">
        <position>966,190,863,1766</position>
        <content>    枕峰寺            乍歸九首</content>
      </line>
      <line id="3">
        <position>863,190,760,1766</position>
        <content>    題坡公贈鄭介夫詩三首</content>
      </line>
      <line id="4">
        <position>760,190,656,1766</position>
        <content>  卷第七┳詩┻</content>
      </line>
      <line id="5">
        <position>656,190,553,1766</position>
        <content>    蔡忠惠家觀帖      癸未瑞慶節</content>
      </line>
      <line id="6">
        <position>553,190,447,1766</position>
        <content>    有感              挽水心先生二首</content>
      </line>
      <line id="7">
        <position>447,190,343,1766</position>
        <content>    林容州別墅        哭林山人</content>
      </line>
      <line id="8">
        <position>343,190,240,1766</position>
        <content>    挽方孚若二首      挽方武成二首</content>
      </line>
      <line id="9">
        <position>240,190,137,1766</position>
        <content>    福州道山          建州</content>
      </line>
      <line id="10">
        <position>137,190,33,1766</position>
        <content>    起來              入浙</content>
      </line>
    </text>
    <annotate count="0" />
  </data>
</page>
```

以上是一个 xml 数据格式的例子。

第 03-05 行，工序记录信息，与全文无关；

第 07-14 行，相应的原扫描图像的总体信息；

第 15-39 行，用于版式还原成 pdf 的排版系数；

第 41-85 行，页内的全文信息，其中：

第 42-83 行为正文信息，包含在<text>中；

一列文字信息包含在一个<line>中；

列在原图像上的位置包含在一个<position>中，逗号隔开的四个数依次表示该列位置的对角坐标的 x 和 y 值；

列内的全文信息包含在一个<content>中；

如果该页有批注信息，将包含在第 84 行的<annotate>中，格式与正文的<line>相同。

下面着重说明<content>中的数据格式。

因为 xml 数据中的全文信息是含版式还原的格式描述的，所以<content>中除文字以外还会出现一些排版符（如第 57 行）。

因排版需要，会出现比较多的符号，如果仅仅关心文字数据的话，可将符号归类为以下几种。

- 大字（正文）

普通大字没有标示

“【” “】” 起始-结束

“［” “］” 起始-结束

- 小字（注释）

“『”　“』”　起始-结束

“┳”　“┻”　起始-结束

“----”小字或小小字在列内的折行符号

- 小小字（注释的注释）

“＃”　“％”　起始-结束

- 插入字

“＊”　“＆”　起始-结束

“＠”　“＆”起始-结束

- 可直接滤掉的排版符

“《”　“》”字带方框

“※”反白字　起始-结束

“‖”字边带方框　起始-结束

“¦”字边带三点　起始-结束

“｜”字边带单线　起始-结束

“h”黑体字　起始-结束

“〖”　“〗”字号+2

“Ⅹ”删除标记　起始-结束

“→”字符横向右旋转 90 度

“≮”　“≯”横排

“∈”空过行高一半的高度

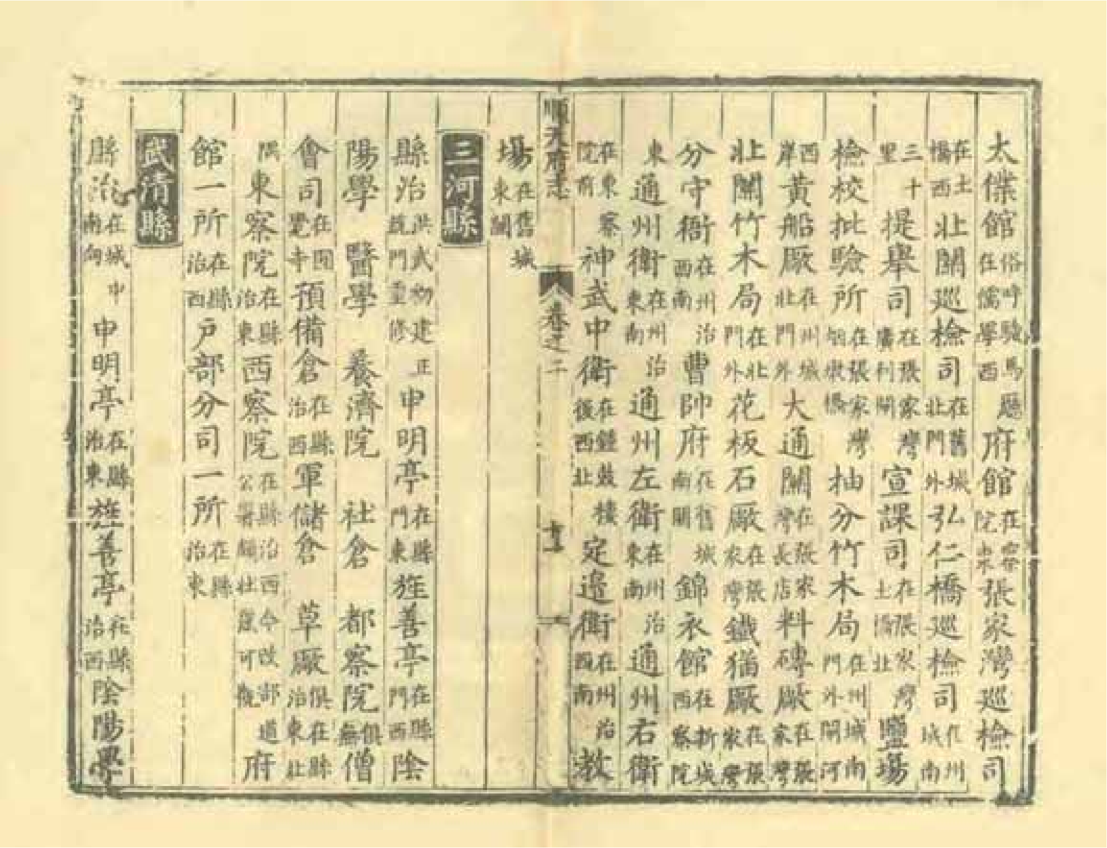
“╧”分段对齐符号

“ ⊙ ”其后的字外带圈

“ J ”句号、“ N ”顿号、“ D ”逗号、“ M ”冒号

可以看出，绝大多数符号像括号一样是一开一关的。标示大、小字的符号对全文数据的提取是最有用的。

（3）全文图像版式



大傑館俗呼驗馬廳在儒學西府館在察院東張家灣巡檢司

在土橋西北關巡檢司在舊城北門外弘仁橋巡檢司在州城南

三十里提舉司在張家灣舊利關宣課司在張家灣土橋北鹽場

檢校批驗所在張家灣舶秋橋抽分竹木局在州城南門外開河

西岸黃船廠在州城北門外大通關在張家灣長店料磚廠在張家灣

北關竹木局在北門外花板石廠在張家灣鐵貓廠在張家灣

分守衙在州治西南曹帥府在舊城南關錦衣館在新城察院西

東通州衛在州治東南通州左衛在州治東南通州右衛

在東察院府神武中衛在鐘鼓樓儀西北定邊衛在州治西南教

順天府志 卷之二 十三

場在舊城東關

三河縣

縣治洪武初建正統間重修中明亭在縣門東旌善亭在縣門西陰

陽學 醫學 養濟院 社倉 都察院俱無僧

會司在圓覺寺預備倉在縣治西軍儲倉 草廠俱在縣治東北

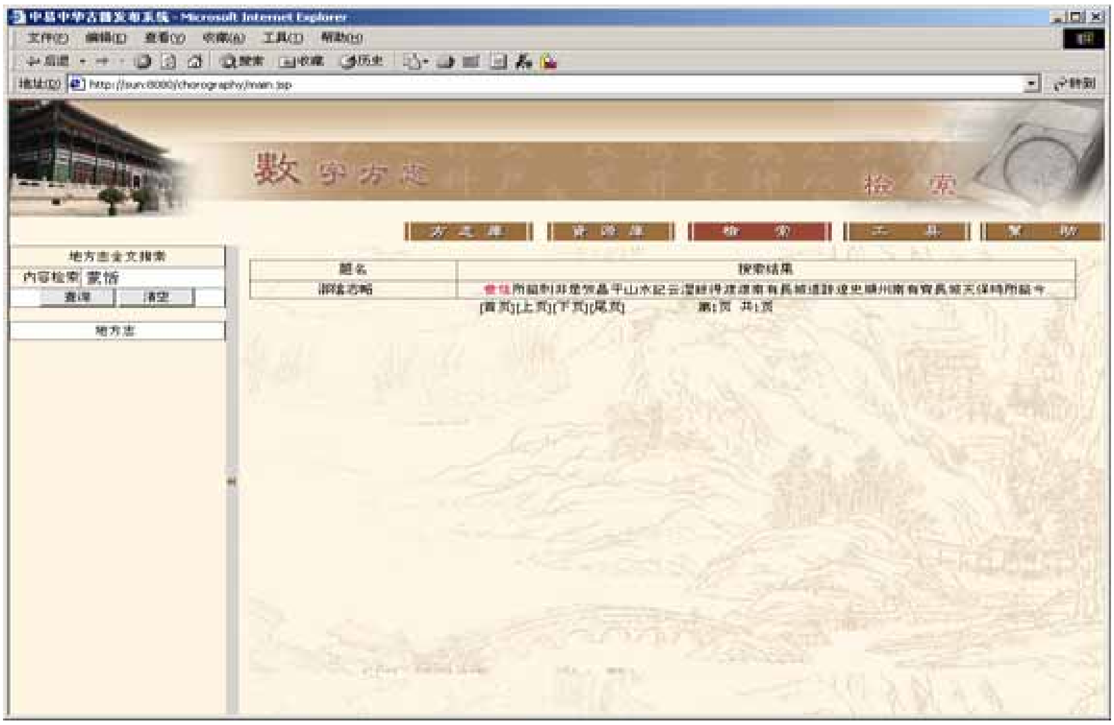
院東察院在縣治東西察院在縣治西今改部道公署舊社倉可廢府

館一所在縣治西戶部分司一所在縣治東

武清縣

縣治在城中南向中明亭在縣治東旌善亭在縣治西陰陽學

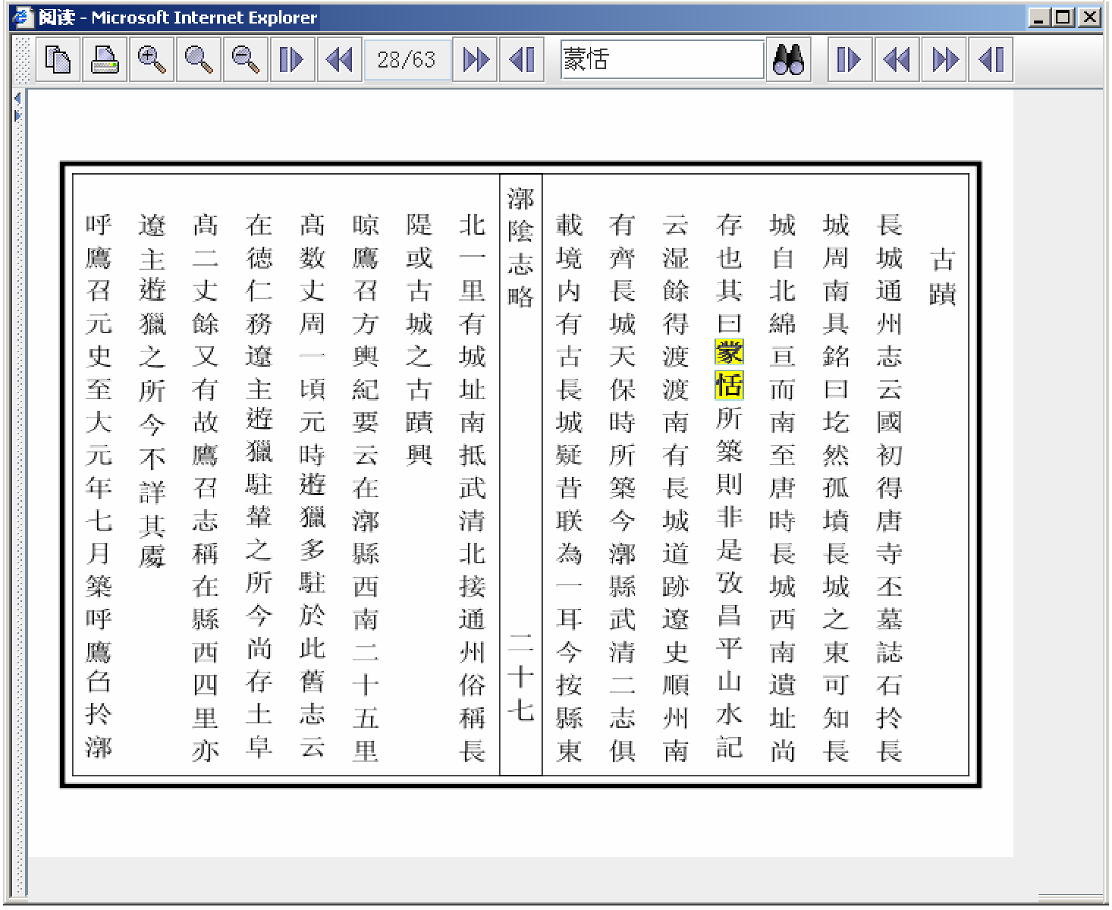
（4）检索


中易中华古籍发布系统 - Microsoft Internet Explorer
文件(F) 编辑(E) 查看(V) 收藏(A) 工具(T) 帮助(H)
地址(D) http://sun:8080/chorography/main.jsp
转到
数字方志
检索
方志库
资源库
检索
工具
帮助
地方志全文搜索
内容检索
蒙恬
查询
清空
地方志

題名	搜索結果
漷陰志略	蒙恬所築則非是孜昌平山水記云溼餘得渡渡南有長城道跡遼史順州南有齊長城天保時所築今

[首页][上页][下页][尾页] 第1页 共1页

（5）全文版式



古蹟

長城通州志云國初得唐寺丕墓誌石抙長城周南具銘曰圪然孤墳長城之東可知長城自北綿亘而南至唐時長城西南遺址尚存也其曰蒙恬所築則非是孜昌平山水記云溼餘得渡渡南有長城道跡遼史順州南有齊長城天保時所築今漷縣武清二志俱載境內有古長城疑昔联為一耳今按縣東

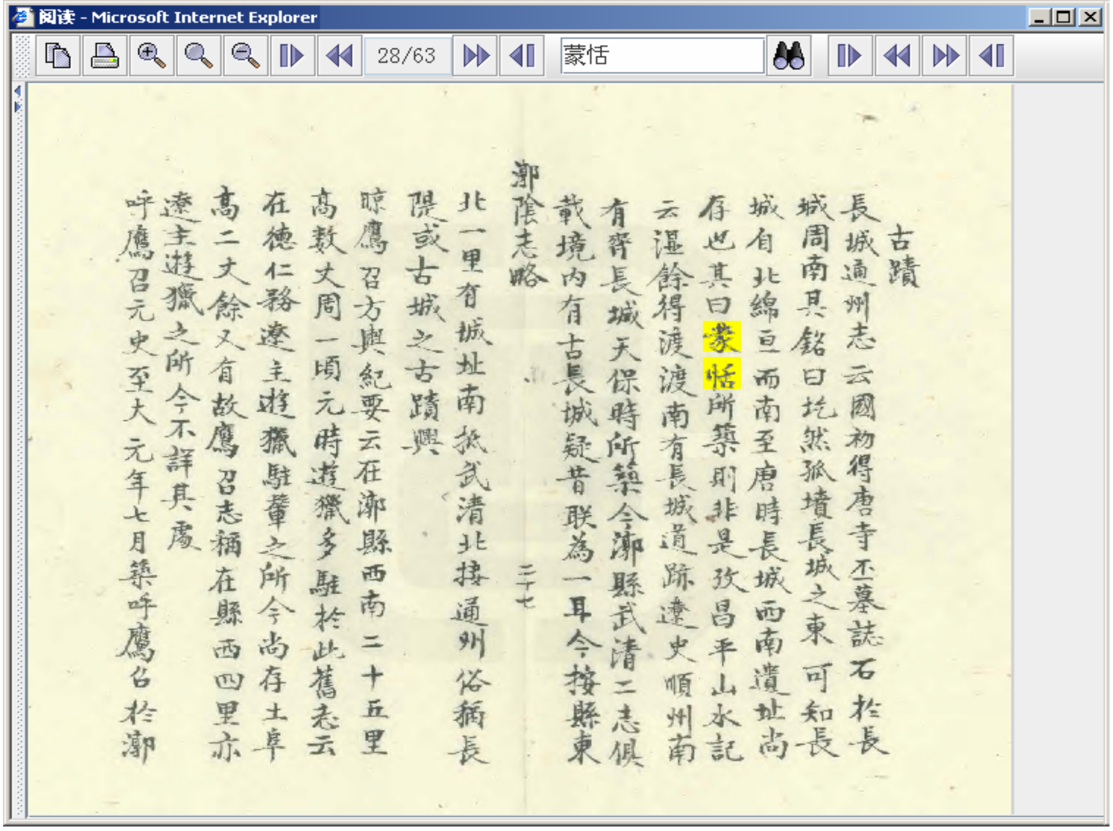
漷陰志略　二十七

北一里有城址南抵武清北接通州俗稱長隄或古城之古蹟興

晾鷹召方輿紀要云在漷縣西南二十五里高数丈周一頃元時遊獵多駐於此舊志云在德仁務遼主遊獵駐蹕之所今尚存土阜高二丈餘又有故鷹召志稱在縣西四里亦遼主遊獵之所今不詳其處

呼鷹召元史至大元年七月築呼鷹台抙漷

（6）全文图像



古蹟

長城通州志云國初得唐寺丞墓誌石於長城周南其銘曰圪然孤墳長城之東可知長城自北綿亘而南至唐時長城西南遺址尚存也其曰蒙恬所築則非是孜昌平山水記云温餘得渡渡南有長城遺跡遼史順州南有齊長城天保時所築今漷縣武清二志俱載境内有古長城疑昔联為一耳今按縣東

漷陰志略　二十七

北一里有城址南抵武清北接通州俗稱長隄或古城之古蹟與

晾鷹台方輿紀要云在漷縣西南二十五里高數丈周一頃元時遊獵多駐於此舊志云在德仁務遼主遊獵駐輦之所今尚存土阜高二丈餘又有故鷹台志稱在縣西四里亦遼主遊獵之所今不詳其處

呼鷹台元史至大元年七月築呼鷹台於漷

致谢：

本文的完成，引用了中国国家图书馆分馆的《数字方正》项目的数据和得到了中国国家图书馆分馆的帮助。引用了中国国家图书馆的《敦煌研究》、《西夏碎金》和《碑帖菁华》项目的数据和得到中国国家图书馆善本特藏部的帮助。相关的技术得到来自于北京中易中标电子信息技术有限公司、北京国图数字技术有限公司、台湾易符智慧科技公司的帮助。借此文正式发表之际，对于帮助者一起致谢。

From Text to Information -- Small Steps towards a Knowledgebase of Tang Civilization

Christian Wittern

1. Introduction

The Knowledgebase of Tang Civilization is aiming at providing a comprehensive interlinked research tool to aid in any research that has some relation to Tang Civilization. To build such a Knowledgebase, work has started with a focus on historical texts, e.g. the two dynastic "standard" histories *Jiu Tang Shu* 舊唐書 by Liu Xu 劉昫 and others (945), *Xin Tang Shu* 新唐書 by Ouyang Xiu 歐陽修 and others (1060) as well as Sima Guang's 司馬光 *Zizhi Tongjian* 資治通鑑, the latter provides a convenient chronicle of events of the Tang years for our purpose.

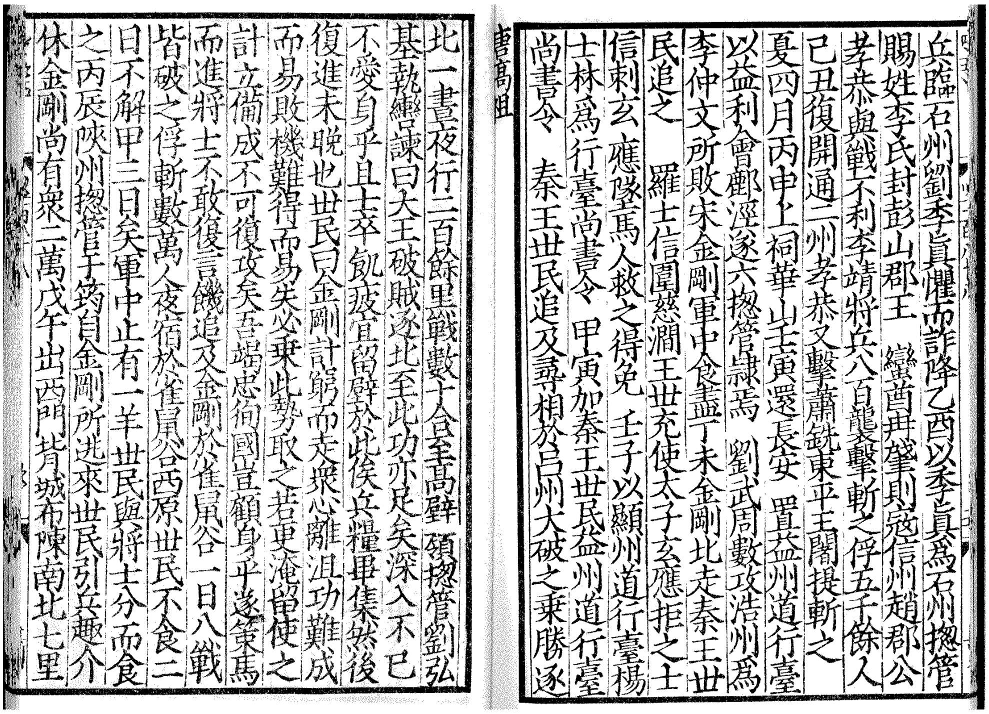
The first challenge in the attempt to make texts available for such a purpose is to transfer them in a digital format that can be used as a fundament for all further work. This transformation will need to reflect the basic structure of the text as well as provide a means to relate to the higher level semantic entities contained therein. With such a text in place, the next step is to isolate, analyze and normalize the information atoms the text relates, which are in this case names of persons mentioned in the texts, titles of books, temporal and spatial identifications (names of places, dates etc) but also administrative titles and acts and finally the events that these text are reporting.

In order to make all this useable in the Knowledgebase, however there is one further step needed: The individual information items need to be related to each other explicitly in some way. While this connection is of course contained in the textual sources, sometimes explicit, but at least impicit, it requires considerable effort to encode these relations in machine-readable form.

2. The *Zizhi Tongjian*

2.1. The text and its tradition

The *Zizhi Tongjian* ‘Comprehensive mirror to aid in government’ (1086) by Sima Guang 司馬光(1019-1086)is easily the most influential single work of chronological (編年體) historical writing in China, in influence on later works second only to the *Shiji* 史記 ‘Record of the Historian’ by Sima Qian 司馬遷. The work has been frequently annotated, excerpted, and expanded; there have been critical examinations and a considerable number of sequels. Figure 1 shows the text available in the widely used Baina 百納 edition. As can be seen, the text is running through, with only occasional spaces to indicate a change of topic.



兵臨石州劉季眞懼而詐降乙酉以季眞爲石州總管賜姓李氏封彭山郡王　蠻酋冉肇則寇信州趙郡公孝恭與戰不利李靖將兵八百襲擊斬之俘五千餘人己丑復開通二州孝恭又擊蕭銑東平王闍提斬之夏四月丙申上祠華山壬寅還長安　置益州道行臺以益利會鄜涇遂六總管隸焉　劉武周數攻浩州爲李仲文所敗宋金剛軍中食盡丁未金剛北走秦王世民追之　羅士信圍慈澗王世充使太子玄應拒之士信刺玄應墜馬人救之得免　壬子以顯州道行臺楊士林爲行臺尚書令　甲寅加秦王世民益州道行臺尚書令　秦王世民追及尋相於呂州大破之乘勝逐

唐高祖

北一晝夜行二百餘里戰數十合至高壁嶺總管劉弘基執轡諫曰大王破賊逐北至此功亦足矣深入不已不愛身乎且士卒飢疲宜留壁於此俟兵糧畢集然後復進未晚也世民曰金剛計窮而走衆心離沮功難成而易敗機難得而易失必乘此勢取之若更淹留使之計立備成不可復攻矣吾竭忠徇國豈顧身乎遂策馬而進將士不敢復言饑追及金剛於雀鼠谷一日八戰皆破之俘斬數萬人夜宿於雀鼠谷西原世民不食二日不解甲三日矣軍中止有一羊世民與將士分而食之丙辰陜州總管于筠自金剛所逃來世民引兵趣介休金剛尚有衆二萬戊午出西門背城布陳南北七里

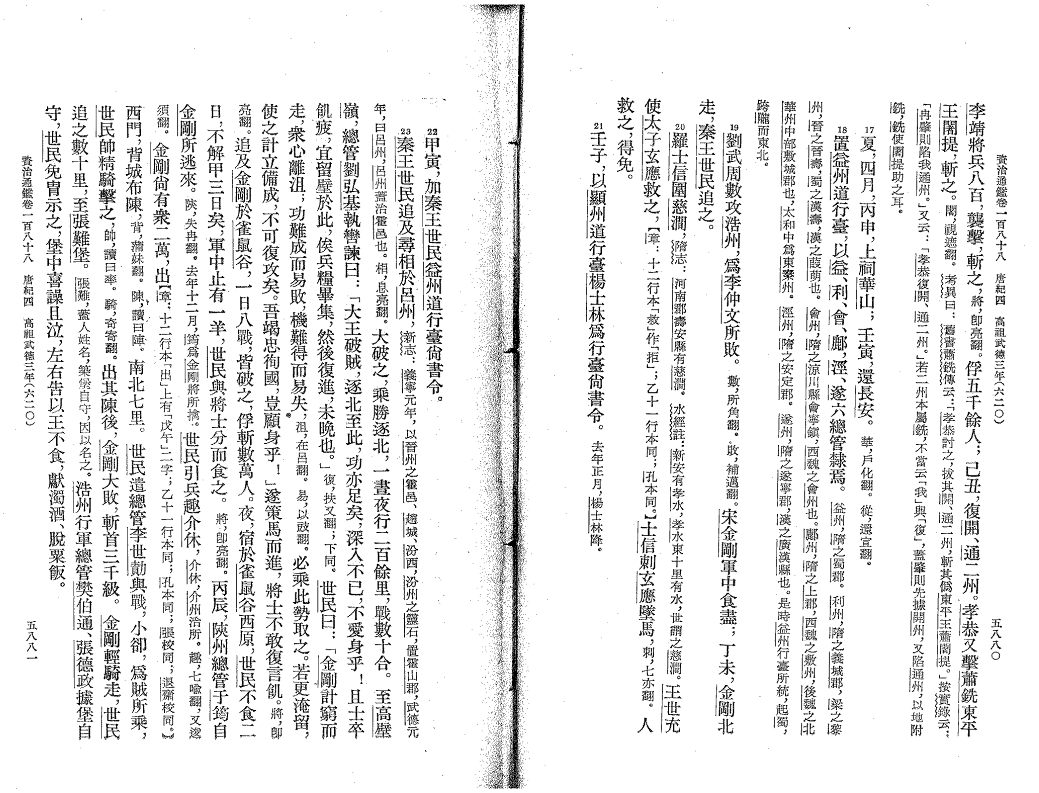
Figure1.

Sima Guang and his compilers[1] made a critical examination of the sources available to them and recorded instances where they had to choose from conflicting accounts. This was published separately as *Zizhi Tongjian Kaoyi* 資治通鑑考異 in 30 juan by Sima Guang himself. Among the annotations that where made to the text of the 資治通鑑 those by Shi Zhao 史炤 (1100-1160)[2] and Hu Sanxing 胡三省 (1230-1287) [3]are the most frequently used. Over time, they have been folded into the text, together with Sima Guangs *Kaoyi* to make up the text, as it is now most conveniently accessed through the punctuated edition from Zhonghua Chuju 中華書局. This text, which has also been the basis for our digitization, is shown in Figure 2.

[1] Apart from Sima Guang himself the three main editors where Liu Ban 劉攽(1023-1089), who was responsible for the Warring States, Qin and Han periods , Liu Shu 劉恕(1032-1078), who took charge of the period from the Three Kingdoms to Sui and Fan Zuyu 范祖禹(1041-1098) who edited the records for the Tang and Five Dynasties period.

[2] The *Shiwen* 釋文 commentary.

[3] 廣註 *Guangzhu* and 音註 *Yinzhu* commentary in 30 juan



資治通鑑卷一百八十八　唐紀四　高祖武德三年（六二〇）　五八八〇

李靖將兵八百，襲擊，斬之，將，卽亮翻。俘五千餘人；己丑，復開、通二州。孝恭又擊蕭銑東平王闍提，斬之。闍，視遮翻。考異曰：舊書蕭銑傳云：「孝恭討之，拔其開、通二州，斬其僞東平王蕭闍提。」按實錄云：「冉肇則陷我通州。」又云：「孝恭復開、通二州。」若二州本屬銑，不當云「我」與「復」，蓋肇則先據開州，又陷通州，以地附銑，銑使闍提助之耳。

17夏，四月，丙申，上祠華山；壬寅，還長安。華，戶化翻。從，還宣翻。

18置益州道行臺，以益、利、會、鄜、涇、遂六總管隸焉。益州，隋之蜀郡。利州，隋之義城郡，梁之黎州，晉之晉壽，蜀之漢壽，漢之葭萌也。會州，隋之涼川縣會寧鎭，西魏之會州也。鄜州，隋之上郡，西魏之敷州，後魏之北華州中部敷城郡也，太和中爲東秦州。涇州，隋之安定郡。遂州，隋之遂寧郡，漢之廣漢縣也。是時益州行臺所統，起蜀，跨隴而東北。

19劉武周數攻浩州，爲李仲文所敗。數，所角翻。敗，補邁翻。宋金剛軍中食盡；丁未，金剛北走，秦王世民追之。

20羅士信圍慈澗，隋志：河南郡壽安縣有慈澗。水經註：新安有孝水，孝水東十里有水，世謂之慈澗。王世充使太子玄應救之，【章：十二行本「救」作「拒」；乙十一行本同；孔本同。】士信刺玄應墜馬，刺，七亦翻。人救之，得免。

21壬子，以顯州道行臺楊士林爲行臺尙書令。去年正月，楊士林降。

22甲寅，加秦王世民益州道行臺尙書令。

23秦王世民追及尋相於呂州，新志：義寧元年，以晉州之霍邑、趙城、汾西，汾州之靈石，置霍山郡，武德元年，曰呂州，呂州蓋治霍邑也。相，息亮翻。大破之，乘勝逐北，一晝夜行二百餘里，戰數十合。至高壁嶺，總管劉弘基執轡諫曰：「大王破賊，逐北至此，功亦足矣，深入不已，不愛身乎！且士卒飢疲，宜留壁於此，俟兵糧畢集，然後復進，未晚也。」復，扶又翻；下同。世民曰：「金剛計窮而走，衆心離沮；功難成而易敗，機難得而易失，沮，在呂翻。易，以豉翻。必乘此勢取之。若更淹留，使之計立備成，不可復攻矣。吾竭忠徇國，豈顧身乎！」遂策馬而進，將士不敢復言飢。將，卽亮翻。追及金剛於雀鼠谷，一日八戰，皆破之，俘斬數萬人。夜，宿於雀鼠谷西原，世民不食二日，不解甲三日矣，軍中止有一羊，世民與將士分而食之。將，卽亮翻。丙辰，陝州總管于筠自金剛所逃來。陝，失冉翻。去年十二月，筠爲金剛將所擒。世民引兵趣介休，介休，介州治所。趣，七喩翻，又逡須翻。金剛尙有衆二萬，出【章：十二行本「出」上有「戊午」二字；乙十一行本同；孔本同；張校同；退齋校同。】西門，背城布陳，背，蒲妹翻。陳，讀曰陣。南北七里。世民遣總管李世勣與戰，小卻，爲賊所乘，世民帥精騎擊之，帥，讀曰率。騎，奇寄翻。出其陳後，金剛大敗，斬首三千級。金剛輕騎走，世民追之數十里，至張難堡。張難，蓋人姓名，築堡自守，因以名之。浩州行軍總管樊伯通、張德政據堡自守，世民免冑示之，堡中喜譟且泣，左右告以王不食，獻濁酒、脫粟飯。

資治通鑑卷一百八十八　唐紀四　高祖武德三年（六二〇）　五八八一

Figure2.

2.2. Structural features of the text

As can be seen from Figure 2, the structure of this text is much more complex than the one seen in Figure 1. Using this as a basis for digitization seems a much more complex and unnecessary cumbersome undertaking. It has however the advantage to provide additional information about the text and the entities the text talks about; for this reason in a long-term perspective, it seemed most convenient to start out from the modern edition.

The text as it is published is divided into the records of 16 dynasties or kingdoms. Of these, the Tang Records 唐紀 in 81 juan is by far the most voluminous. For the Tang Knowledgebase, work did concentrate exclusively on this part of the text. Within the Tang Records, the division is by Emperor, then by era period and finally by year. The editors choose to treat a calendar year as the basic unit, although obviously some changes of emperor or era do not fall at the beginning of the year. Within a year, the narrative is divided into individual paragraphs, which mostly follow those white spaces of the earlier editions, as can be seen by comparing the ‘Zhonghua Shuju’ edition with the ‘Baina’ edition. In cases where the editors did feel a further division was necessary, they did introduce their own paragraphs and, to distinguish these two types of paragraphs, the ones that appeared already in the early woodblock edition (and might have been introduced for all we know by Sima Guang himself) have numbers attached to them, which run through for one whole year. In a first approximation, such a paragraph can be seen as the most basic narrative unit, reporting one ‘event’.

Within a paragraph, the notes of various types are distinguished from the running text by smaller point size, but they are not further separated neither by source nor by category. Other clearly distinguishable features of the text are waivy and straight lines, which indicate titles of texts and other named entities respectively.

2.3. Translating structure to markup

As described above, the structural features of the text visible in the layout point to semantic features. For the electronic text to be useful, these features have to be named and transposed into machine readable form. For the purpose of this project, the *Guidelines for Electronic Text Encoding and Interchange*[4] have been used and adapted to the special purposes of this project[5]. In most cases the transposition from structure to markup has been straightforward and could be applied in a semi-automated fashion. What proved to be most difficult was the introduction of distinctions that had no equivalent in the printed page. For example, the named entities marked with a straight line lumped together quite distinct features:

- Personal names (<rm>)[6]
- Names of Places (<dm>)
- Era names (<y>)
- Names of dynastic periods or kingdoms (<dyn>)

Identifying and assigning the proper categories for these items proved to be by far the most time consuming task. The difficulty here is that even with supporting authority files, which we gradually created during this project, the assignment itself still has to be checked carefully. Another problem was the desire to not only identify the categories, but also link to the authority file, which might have different names for a person and additional information.

The text in smaller print, which lumped together all annotations that had been added to the original text either by Sima Guang or by a range of later commentators and editors, can be distinguished in the following way:

- text critical notes (<app>)

[4] Edited by Michael Sperberg-McQueen and Lou Burnard. First published in 1994, the most recent revision is of 2002.

[5] Adaptability is one of the major features of these Guidelines, which allow them to be both very broadly applicable and yet specific enough for varying local requirements. The adoption in this case consisted mostly of removing unnecessary elements and making the content models more stringent, so that it was easier to check the markup.

[6] In parentheses the elements used to indicate this feature. See the Appendix for a correspondence of these notations to the standard TEI notation.

- sound glosses (<gloss>)
- cross references (<xref>)
- critic of sources (this is indicated by a reference to *Zizhi Tongjian Kaoyi*))
- additional information (provided by Hu Sanxing or others)
- citations (<cit>)

It is by no means always obvious, who is responsible for a given note and when it was introduced, although in some cases, the name is mentioned explicitly.

In applying the markup, the need to economize on the amount of effort that is required to create texts with markup adds additional limitations; for this reason not all desirable features have been introduced.

A short excerpt from the markup of the master files, showing a part from the text in Figure 2 is given in the Appendix.

3. Beyond markup

The purpose of applying this type of markup is of course to make available the text content for further processing. Within the context of the Knowledgebase, this means deriving information from the text, abstracting from the individual expressions used there.

One area where this is specifically important is with regard to the persons mentioned in the text. Depending on the context and their standing in the world, many persons are referred to by different names in the course of their life, and this is naturally also reflected in our sources. Additionally, the chronological account and the several layers of notes to refer to persons differently in many cases.

In addition to simply marking textual features as being names of places, persons and so on, work has began to identify persons and relate their names; this is done separately from the text that constitutes the ‘resource layer’ in what we call the ‘information layer’. This is of course a significant task that even for a text of comparatively limited scope, as the Tang Records of the 資治通鑑, where for about the first half of the Tang period[7], at the moment we have identified more than 6000 different persons mentioned in the text[8]. This identification is complicated by the fact that names or identifications such as Prince of Qin (秦王) are used for different persons at different times; frequently persons are only mentioned by

[7] To be exact, at present the period from the first year of Wude 武德 (618) to the end of Tianbao 天寶(756) has been analyzed.

[8] This figure, which should of course not be taken to be the final toll, does also include persons from earlier periods that are mentioned in the text, as well as the names of later commentators and other persons, whose names appear in the sources. Only very preliminary work has been done to further analyze and categorize these data so far and we expect to devote more time and effort on this aspect in the years to come.

their given name, which may increase the number of identical strings that still have to be assigned to different records.

While the amount of effort to identify persons in such a way is prohibiting high, on the other side, we hope that the resulting data can be used as a training corpus to further annotate texts in this way semi-automatically, both texts from the Tang period, where the data and its content in the information layer can be straightforwardly applied, but also texts from other periods, where heuristics of the textual context derived from these data can provide hints to what has to be considered a personal name.

To give an example, here is a list of names used in our sources for the second Tang emperor Taizong, who is by far most frequently mentioned in the period considered here.

- 世民 (237[9])
- 唐太宗 (4)
- 太宗 (215)
- 太宗文武皇帝 (1)
- 太宗文武聖皇帝 (1)
- 文武 (1)
- 文武皇帝 (1)
- 文武大聖大廣孝皇帝 (1)
- 文武大聖皇帝 (1)
- 文武皇帝 (1)
- 文皇帝 (2)
- 秦公世民 (1)
- 秦王 (4)
- 秦王世民 (76)
- 趙公 (1)
- 趙公世民 (3)

There are 17 different appellations for Taizong and it should be clear that it would be quite impossible to find all places where he is mentioned in the source text by trying to perform a simple search for his name.

There are of course countless more aspects in which even a comparatively simple markup like the one used here can be used to further extract and analyze the text to gain new information. It is for example interesting to see at which time which name was used to refer to a person, or which nominal title was conferred

[9] The figures in parentheses are the number of occurrences in the text of the *Zizhi Tongjian* from 618 up to the end of the era Tianbao (756).

at which time. Since our text is a chronological record, it is quite easy to derive this information from the structure of the text.[10] The following Table 1 gives the names as they have been used, broken down by year; it is also mentioned whether the name is mentioned in the narrative main text, or somewhere in the commentary.[11]

Table1.

Year	Occ. total	Occ. in text (note); total: 556
618 (武德−1)	36	• 秦公世民 1 (0) • 世民 19 (1) • 趙公 1 (0) • 趙公世民 3 (0) • 太宗 0 (2) • 秦王 2 (0) • 秦王世民 7 (0)
619 (武德−2)	23	• 秦王世民 9 (0) • 太宗 0 (1) • 世民 11 (2)
620 (武德−3)	58	• 秦王世民 18 (0) • 世民 27 (0) • 太宗 1 (13)
621 (武德−4)	85	• 秦王世民 16 (0) • 太宗 1 (11) • 世民 57 (1)
622 (武德−5)	66	• 秦王世民 14 (0) • 世民 35 (1) • 太宗 4 (16)
623 (武德−6)	6	• 秦王世民 5 (0)

[10] Since the text does give exact dates for many events, we do in fact plan to make this information accessible in machine readable form, but it has not yet been applied to the master copy of our text. It should also be noted that the narrative is at places not strictly chronological, but does follow the events where necessary, so the purely positional computation does have its limitations.

[11] Since this table is just ment to illustrate the idea, some years between the beginning of Zhenguan 貞觀 and the end of Tianbao have been skipped to save some space.

Year	Occ. total	Occ. in text (note); total: 556
		• 世民 1 (0)
624 (武德-7)	46	• 太宗 2 (7) • 秦王世民 5 (0) • 世民 32 (1)
625 (武德-8)	1	• 秦王世民 1 (0)
626 (武德-9)	57	• 秦王世民 1 (0) • 世民 46 (3) • 太宗 4 (3)
627 (貞觀-1)	7	• 太宗 0 (5) • 秦王 1 (1)
628 (貞觀-2)	7	• 太宗 1 (6)
649 (天寶-8)	2	• 太宗 1 (0) • 文武大聖皇帝 1 (0)
754 (天寶-13)	2	• 太宗 1 (0) • 文武大聖大廣孝皇帝 1 (0)

4. Conclusions

It should be clear that this is just one example of how by abstracting from information contained in the text new layers of information can be constructed. While only a very simple example has been given, other rather easily gained information items are the co-occurrence of persons and places participating in an event (as described in one paragraph). Also, formal records of astronomical events, sacrifices and excursions of the Emperor can rather easily be harvested into machine readable form. At times, this might resemble approaches common from data mining methodology, there are nevertheless differences. Since the research target is information to be derived from narrative texts, all irregularities and inconsistencies of natural language and literary style have to be taken into account. Data mining and analytic tools are thus most usefully employed to give a rough dataset, which is then refined by a human researcher.

For this reason, the goal of constructing the *Knowledgebase of Tang Civilization* does not primary lie in constructing a comprehensive repository of information and resources pertaining to the Tang, but to provide the researcher with an infrastructure that assists her in getting answers to her questions from the sources, and sometimes even getting answer for questions she did not ask.

5. References

C. M. Sperberg-McQueen Lou Burnard *Guidelines for Electronic Text Encoding and Interchange P4*, TEI Consortium Oxford, Providence, Charlottesville, Bergen, 2002.

Biaodian Zizhi Tongjian Xiaozu 標點資治鑑小組*Zizhi Tongjian* 資治通鑑, Beijing 1956.

6. Appendix

6.1. Markup Example

The following is an example of how a part of the text in the sections shown in Figure 1 and 2 has been encoded in our files. The section startes with the header for the year, but then skips right directly to paragraph 17 on page 5880.

The following elements have been abbreviated: <dm> for <name type="place">, <rm> for <name type="person">, <y> for <date>, <dyn> for <name type="dynasty">. Instead of linebreaks, <lb> indicates places where breaks occur in our internal master documents.

<div><headn="武德-3">三年<noteplace="inline">(庚辰 、六二０)</note></head>

(...)

<div n="17"><p>夏，四月，丙申，上祠<dm>華山</dm>；壬寅，還<dm>長安</> 。<lb n="5880-175"/><note place="inline">華，戶化翻 。從，還宣翻 。</note></p></div>

<divn="18"><p>置<dm>益州道</dm>行臺，以<dm>益</dm> 、<dm>利</dm> 、<dm><lbn="5880-200"/>會</dm> 、<dm>鄜</dm>，<dm>涇</dm> 、<dm>遂</dm>六總管隸焉 。<noteplace="inline"><dm>益州</dm>，<dynkey="ch174">隋</dyn>之<dm>蜀郡</dm> 。<dm>利州</dm>，<dynkey="ch174">隋</dyn><lb n="5880-225"/>之<dm>義城郡</dm>，<dynkey="ch166">梁</dyn>之<dm>黎州</dm>，<dynkey="ch110">晉</dyn>之<dm>晉壽</dm>，<name>蜀</name>之<dm>漢壽</dm>，<dynkey="ch129">漢</dyn>之<dm>葭萌</dm>也 。<lb n="5880-250"/><dm>會州</dm>，<dynkey="ch174">隋</dyn>之<dm>涼川縣</dm><dm>會寧鎮</dm>，<name>西魏</name>之<dm>會州</dm>也 。<dm>鄜州</dm>，<dynkey="ch174">隋</dyn>之<dm><lbn="5880-275"/>上郡</dm>，<name>西魏</name>之<dm>敷州</dm>，<dynkey="ch169">後魏</dyn>之<dm>北華州</dm><dm>中部</dm><dm>敷城郡</dm>也，<y to="中" value="太和">太和中</y><lbn="5880-300"/>為<dm>東秦州</dm> 。<dm>涇州</dm>，<dyn key="ch174">隋</dyn>之<dm>安定郡</dm> 。<dm>遂州</dm>，<dyn key="ch174">隋</dyn>之<dm>遂寧郡</dm>，<dynkey="ch129">漢</dyn>之

<dm><lbn="5880-325"/>廣漢縣</dm>也。是時<dm>益州</dm>行臺所統，起
<name>蜀</name>，跨<dm>隴</dm>而東北。</note></p></div>
<divn="19"><p><rmkey="r00280">劉武<lbn="5880-350"/>周</rm>數攻
<dm>浩州</dm>，為<rmkey="r01611">李仲文</rm>所敗。<note
place="inline">數，所角翻。敗，補邁翻。</note><rmkey="r06179"><lb
n="5880-375"/>宋金剛</rm>軍中食盡；丁未，<rmkey="r06179">金剛</rm>
北走，<rmkey="r01602">秦王世民</rm>追之。</p></div>
<divn="20"><p><rmkey="r03081">羅士<lbn="5880-400"/>信</rm>圍<dm>
慈澗</dm>，<noteplace="inline"><cit><title>隋志</title>：<q><dm>河
南郡</dm><dm>壽安縣</dm>有<mref="c5880-401">慈澗</m>。
</q></cit><cit><title>水經註</title>：<q><dm>新安</dm>有<dm><lb
n="5880-425"/>孝水</dm>，<dm>孝水</dm>東十里有水，世謂之<m
ref="c5880-401">慈澗</m>。</q></cit></note><rmkey="r02590">王世充
</rm>使<rmkey="r04210">太子玄應</rm><lbn="5880-450"/><app
resp="【章】"><lem>救</lem><rdgwit="十二行本；乙十一行；孔本">拒
</rdg></app>之，<lbn="5880-475"/></note><rmkey="r03081">士信</rm>
刺<rmkey="r04210">玄應</rm>墜馬，<noteplace="inline">刺，七亦翻。
</note>人救之，得免。</p></div>
<divn="21"><p>壬子，<lbn="5880-500"/>以<dm>顯州道</dm>行臺<rm
key="r05490">楊士林</rm>為行臺尚書令。<noteplace="inline">去年正
月，<rmkey="r05490">楊士林</rm>降。<lbn="5880-525"/></note><pb
n="5881-"/></p></div>
<divn="22"><p>甲寅，加<rmkey="r01602">秦王世民</rm><dm>益州道
</dm>行臺尚書令。</p></div></div>

多语言处理系统研究[1]

牛振东
中国数字图书馆有限责任公司 北京 100081
zniu@nlc.gov.cn

摘要　本文研究多语言处理系统，重点论述多语言处理系统的体系结构和多语言主题词表等关键技术。

关键词　多语言处理、主体词表、 知识分类、多语言检索系统

The Research and Design of Multilingual Process System

Zhendong Niu
China Digital Library Corp. Ltd. Beijing, China 100081
zniu@nlc.gov.cn

Abstract: This paper discusses the multilingual processing system. It mainly focuses on the infrastructure of multilingual process system and the related key techniques including thesaurus, classification schemes etc.

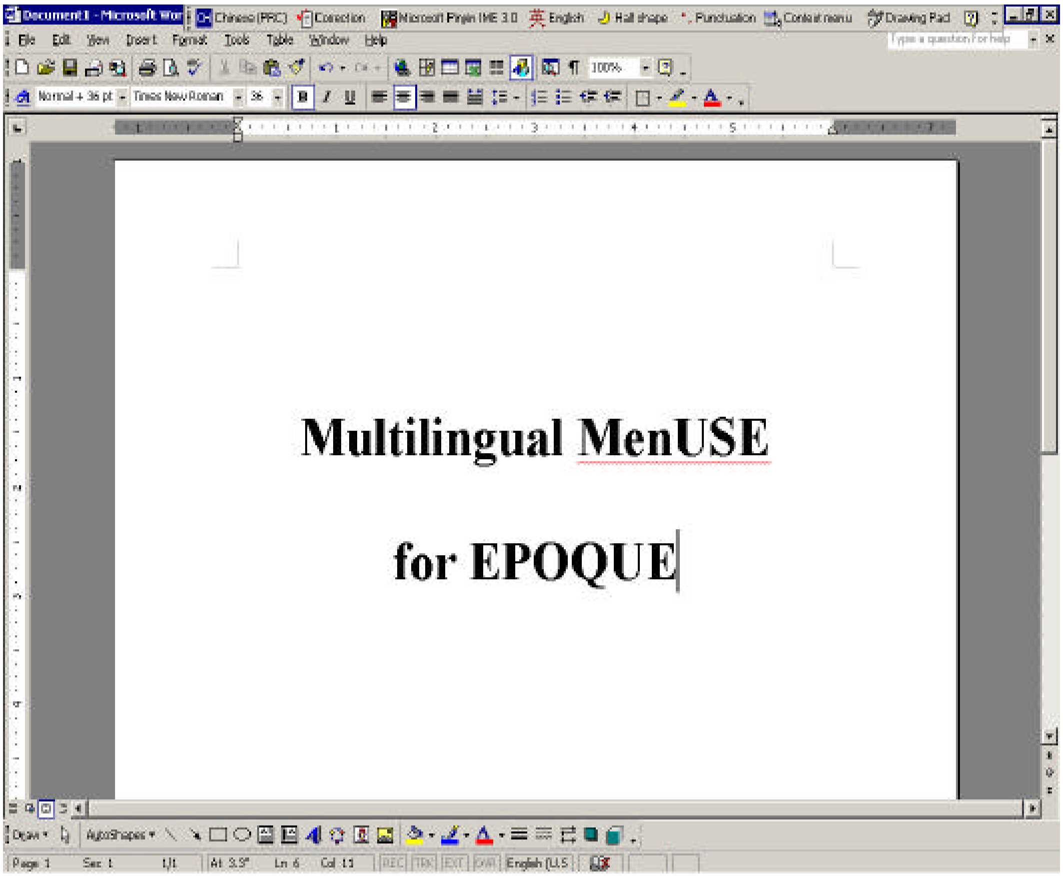
Keywords: multilingual process, thesaurus, classification schemes, multilingual retrieval system

1 引言

多语言信息服务是一个相当复杂的领域，它涉及信息管理、语言学、人工智能、计算机软件等领域的技术。多语言处理系统不仅仅是文字翻译问题，它还包括信息采集、分类、存储、索引、检索和显示输出等多个关键环节。

[1] 本文部分研究得到中国国家图书馆 2004 年度科研项目文化网格项目支持。

在欧共体总部所在地布鲁塞尔，欧洲议会有一个数百人的机构专门来处理多语言信息并提供相关的服务。早在90年代初期，欧洲议会便投入经费研究使用多语言处理系统，图1.1表示了一个用于欧洲议会的多语言处理系统界面。



Multilingual MenUSE

for EPOQUE

图 1.1　用于检索欧洲议会文献库的多语言信息系统

在过去的十年里，中文信息处理和智能接口技术都有了长足的飞跃。但是，由于我国国情和其他原因，我国缺少生产大型多语言信息服务平台方面的经验，缺少大型（多点可切换式）多语言信息服务系统，在多文种信息处理方面，我们更存在很多空白，在小规模双语言的研发上也常常存在着许多局限性。为此，国家“863”高科技研究发展计划，以及国家计委、信息产业部和外经贸委的“振兴软件产业行动计划”中，多语言信息服务系统的开发问题被作为一个重要内容提出。

数字图书馆的一个理想是实现人类在任何地方可以检索世界上任何国家的任何文字的数字资源信息，因此，多语言信息处理系统是数字图书馆领域必须解决的问题。

针对我国现状，结合国际上多语言处理的成果，本文主要研究多语言信息处理和综合服务平台的结构及包括多语言主题词表、自动信息分类和索引编制软件

包、多语言检索引擎等核心组件等在内的多语言处理关键技术，设计一种基于主题词表的多语言用户检索系统。 本第二部分重点介绍多语言信息服务平台系统的结构设计，第三部分介绍多语言处理系统的关键组成部分和关键技术，第四部分对有关问题进行讨论，最后一部分是致谢。

2 多语言信息服务平台的系统结构

多语言信息服务平台涉及信息的采集、分类、编码、存储管理、翻译、索引、检索等方面。一个典型的多语言处理信息服务平台应包括数据信息的采集系统、信息分类系统、多语言搜索引擎系统、多语言索引系统、内容管理系统、用户接口系统等关键部分[1-4]。图2.1 表示了一个多语言信息服务系统的基本结构。

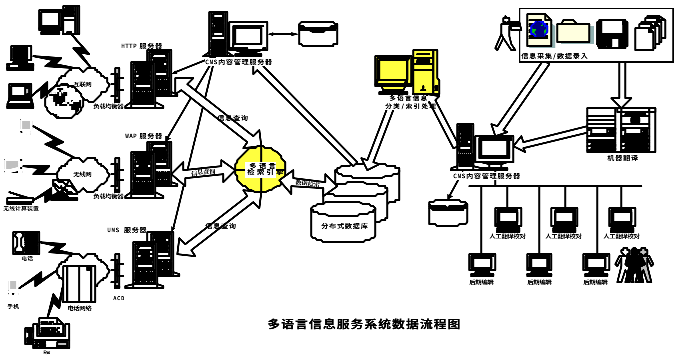


图 2.1 多语言信息服务平台的系统结构

在图 2.1 中，信息采集和数据录入系统获得各种信息资源；多文种信息分类和索引编制软件包是一个专用软件包用来对文章，网页等进行自动扫描，根据关键字来进行自动或半自动分类及存入数据库并对其编制索引等 ;多文种检索引擎是信息检索的核心组件，提供对多语言的检索服务；多语言信息综合服务平台是上述系统和 Web Services，内容管理等相关模块在互联网上的系统集成实现，作为多语言信息应用服务系统，它支持多种用户终端等，具有直观、易用、检索效率高等特点。

3　多语言处理系统的关键技术

本节论述多语言信息服务平台的主要构件和关系，并重点介绍主题词表技术。图3.1 显示出多语言检索引擎、主题词表和多语言分类索引系统组件在多语言信息系统中的重要性和相互关系。

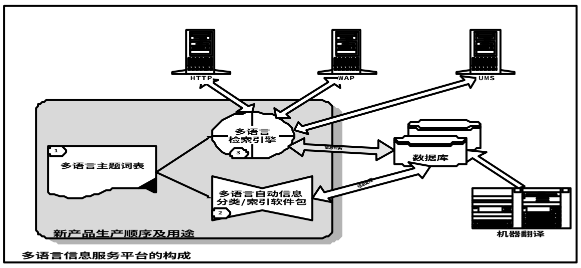


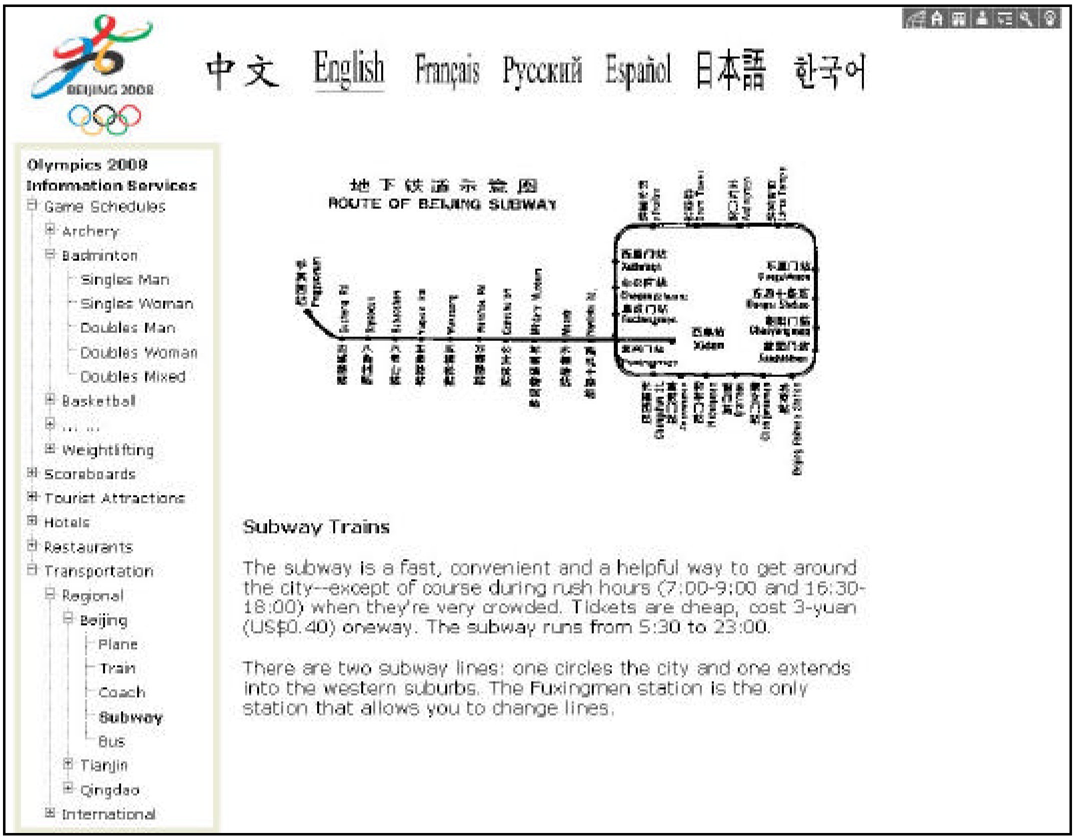
图 3.1 相关组件与多语言服务平台的关系

信息或知识分类结构是建立多语言信息服务系统的基础，只有做好这部分工作，多语言自动分类，自动索引和检索引擎的开发才能有效地完成。多文种主题词表是多语言信息或知识的分类结构表达系统[5-6]。

在图3.1中，多语言主题词表是多语言信息处理的核心，它不仅是研发多语言信息分类和索引编制软件包，及检索引擎的基础，它还可以有来设计直观、容易使用的智能用户界面。

图3.2示范如何用多语言主题词表里设计主题树,方便用户浏览所需信息。例如，一个到北京来的游客想查找本地交通信息时，他可以通过（结构化）树状景观，逐层深入，找到他想要的信息如地铁。 当用户点击地铁图的某一个站点时，其相关区域的信息可进一步显示出来。

除信息分类、用户界面技术外，多语言处理系统的其他重要技术包括中间信息的表示和搜索引擎等关键技术等。目前的主流中间信息表示技术是建立相应标准并提供对XML表示的支持。



Olympics 2008 Information Services

- Game Schedules
  - Archery
  - Badminton
    - Singles Man
    - Singles Woman
    - Doubles Man
    - Doubles Woman
    - Doubles Mixed
  - Basketball
  - … …
  - Weightlifting
- Scoreboards
- Tourist Attractions
- Hotels
- Restaurants
- Transportation
  - Regional
    - Beijing
      - Plane
      - Train
      - Coach
      - Subway
      - Bus
    - Tianjin
    - Qingdao
  - International

Subway Trains

The subway is a fast, convenient and a helpful way to get around the city--except of course during rush hours (7:00-9:00 and 16:30-18:00) when they're very crowded. Tickets are cheap, cost 3-yuan (US$0.40) oneway. The subway runs from 5:30 to 23:00.

There are two subway lines: one circles the city and one extends into the western suburbs. The Fuxingmen station is the only station that allows you to change lines.

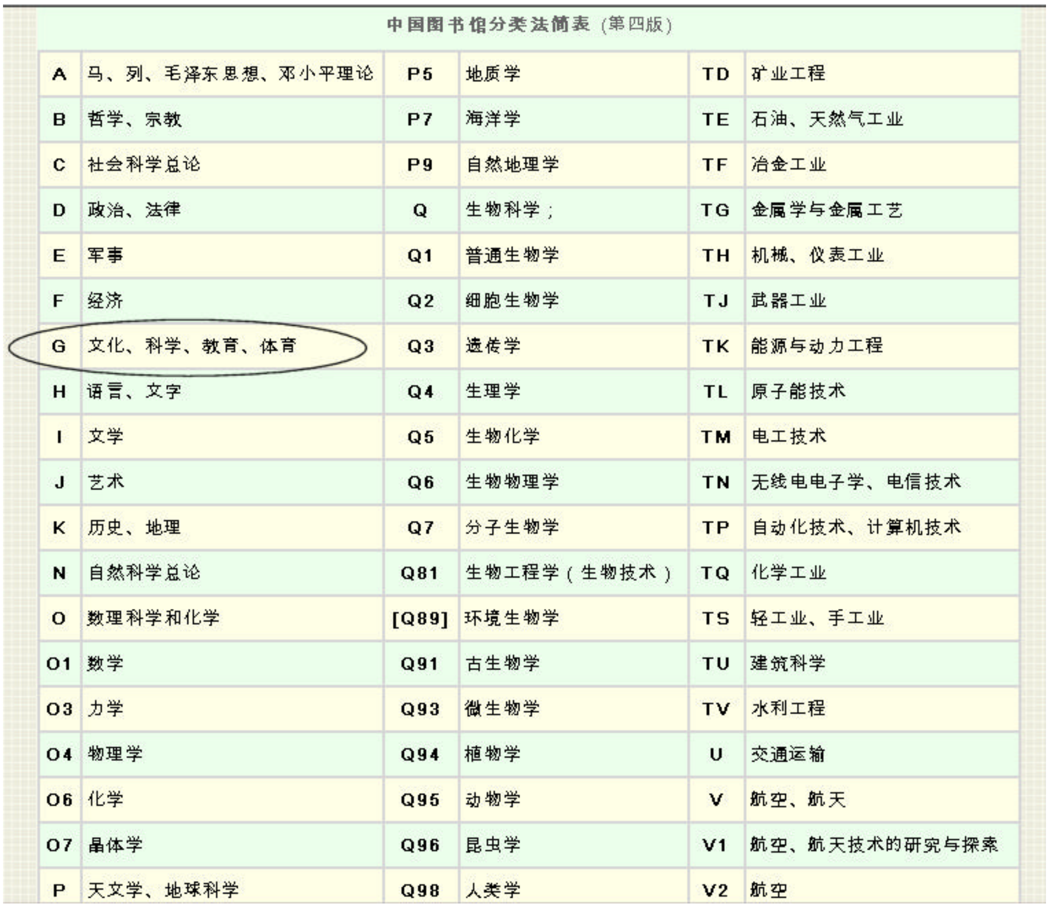
图 3.2 利用主题词表建立起来的树状景观（Tree View）方便用户浏览和查找所需信息

4 讨论

信息或知识分类结构（classification schemes）是信息服务系统的开发和生产的关键。如果没有对信息或知识进行有效的组织和分类，机器翻译后的信息也难以得到充分的使用，用户恐无法找到他们需要的信息。

近年来我国在中文信息分类上作了很大的投入，中国图书馆分类法编委会及40家合作单位共同努力，于1998年在“ 中图法 ”第三版的基础上成功的编制了我国第一部 “ 中国分类主题词表 ”。它为实现机助标引，自动标引，提高检索效率创造了条件。“ 中国分类主题词表 ”在2000年又以“ 中图法 ”第四版为基础进行

了更新。



中国图书馆分类法简表 (第四版)

A	马、列、毛泽东思想、邓小平理论	P5	地质学	TD	矿业工程
B	哲学、宗教	P7	海洋学	TE	石油、天然气工业
C	社会科学总论	P9	自然地理学	TF	冶金工业
D	政治、法律	Q	生物科学；	TG	金属学与金属工艺
E	军事	Q1	普通生物学	TH	机械、仪表工业
F	经济	Q2	细胞生物学	TJ	武器工业
G	文化、科学、教育、体育	Q3	遗传学	TK	能源与动力工程
H	语言、文字	Q4	生理学	TL	原子能技术
I	文学	Q5	生物化学	TM	电工技术
J	艺术	Q6	生物物理学	TN	无线电电子学、电信技术
K	历史、地理	Q7	分子生物学	TP	自动化技术、计算机技术
N	自然科学总论	Q81	生物工程学（生物技术）	TQ	化学工业
O	数理科学和化学	[Q89]	环境生物学	TS	轻工业、手工业
O1	数学	Q91	古生物学	TU	建筑科学
O3	力学	Q93	微生物学	TV	水利工程
O4	物理学	Q94	植物学	U	交通运输
O6	化学	Q95	动物学	V	航空、航天
O7	晶体学	Q96	昆虫学	V1	航空、航天技术的研究与探索
P	天文学、地球科学	Q98	人类学	V2	航空

图表 4.1 中图法信息分类结构

但是，我们的主体分类词表仅对各主题类别进行了代码化，并没有对其进行数码化。面对今天信息技术和网络技术的高速发展，这种处理方式已无法满足计算机高速处理的前提条件。此外，在中图分类法中，对数字资源尤其是网络资源、新闻、体育，旅游和交通等主要方面都存在着收录不足问题，如上表所示，“中图法”仅仅对体育进行了相当粗略的主题分类。

中文的“中国分类主题词表”尚欠完善，在多文种信息分类方面，我国更存在着不少空白点，这种缺陷显然束缚了我国多文种信息服务的发展。多文种信息分类和知识结构（knowledge organization）是计算机信息处理的重要基础，只有搞好这方面的工作，多文种自动分类，自动索引和检索引擎的开发才能有效地

完成。

当前中文的主题词表（thesaurus）和受控词汇表（controlled vocabularies）欠完整，分类码（classification code），或结构化相关主题词表（subject headings）也有很多不足。建设好的多语言处理平台以上问题是关键。

针对上述问题，我们正开发基于本体的知识分类体系[7]，并期望通过引入知识本体来提高多语言处理的效率，整个系统的原型开发也在进行中。

5　致谢

感谢美国Berkeley实验室研究员李春生博士提供材料并提供很多有价值的建议。感谢中国国家图书馆的研究支持。

参考文献：

1. Jerome Yen. Multimodel and multilingual informedia – The iVIEW system. 2nd CCDL. Beijing. 2004.
2. Michael R. Lyn, Edward Yau, Sam Sze. A multilingual, multimodal digital video library system. Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries. P 145-153. ACM Press. 2002
3. Akira Maeda. Multilingual information processing for Digital libraries. Department of Computer Science, Ritsumeikan University 2001
4. Nuno Freire. Integration of Multilingual Classification Systems with the Dienst digital library system. http://www.ercim.org/ws-proceedings/DEL0S8/freire.pdf.
5. A Steven Pollitt. The key role of classification and indexing in view-based searching. IFLA'97 Copenhagen. Aug 31 – Sept. 2. 1997
6. C S Li A S Pollitt & M P Smith. Multilingual MenUSE – A Japanese front-end for searching English language database and vice versa. 14th BCS IRDG information retrieval colloquium, Lancaster 1992 Tony McEnery & Chris Paice (eds). Springer Verlag. Pp 14-37.
7. Zhendong Niu Mingkai Dong Jie Zhang Huaming Chen. A knowledge based solution for Digital libraries. 2nd CCDL. Beijing. 2004.

Character Processing Based on Character Ontology

MORIOKA Tomohiko

1 Introduction

We use characters as a basis for data representation in computers, and as a tool for communication over computer networks. Computer programs are made of characters, we exchange mails that are sequences of characters, and many of the contents available over the Internet are realized in the form of characters.

Currently, in the field of information processing, characters are defined and shared using coded character sets. Character processing based on coded character sets, however, has two problems:

1. Coded character sets do not always contain a necessary character
2. Characters in coded character sets have fixed semantics

To resolve the problems, I proposed "Chaon" model which is a new model of character processing based on character ontology. In Chaon model, characters are defined, represented and processed according to its own character databases. Characters in Chaon model are independent from coded character sets for information interchange, and semantics of the characters stored in the database can be freely added or altered.

To realize the character processing based on Chaon model, I started "CHISE (Character Information Service Environment)" project with some other members. In CHISE project, we have developed some systems and databases to edit/process/print characters and texts. CHISE project is an open source project, so the results are freely distributed. We are realizing character processing environment based on Chaon model. In this paper, I explain an overview of the current state of character processing technology in CHISE project.

2 Chaon model

In Chaon model of character representation, a character is not a code point of a coded character set, but a set of the features it has. Characters are represented as character objects, and character objects are defined by character features. Character objects and character features are stored in character databases, and a character can be accessed using its feature as a key.

There are various information related with characters, so we can regard various things as character features, for example, shapes, phonetic values, semantic values, code points in various character codes. Figure 1 shows a sample image of a character representation in Chaon model which indicate a character "吉".

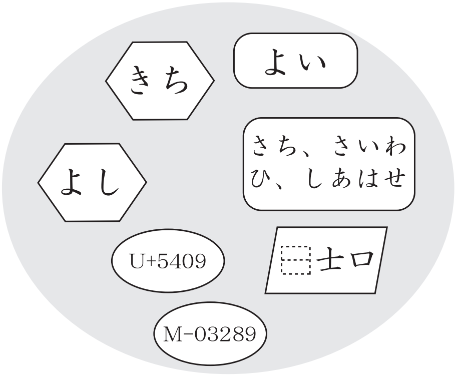


Figure 1: sample of character features to indicate "吉"

In CHISE model, each character is represented by a set of character features, so we can use set operations to compare characters. Figure 2 shows a sample of a Venn diagram of character objects. The diagram indicates that there are common semantic and phonetic values between characters “言”, “云” and “謂” even if they don’t have the same glyph, and characters “云” and “雲” have the common semantic and phonetic values in China.

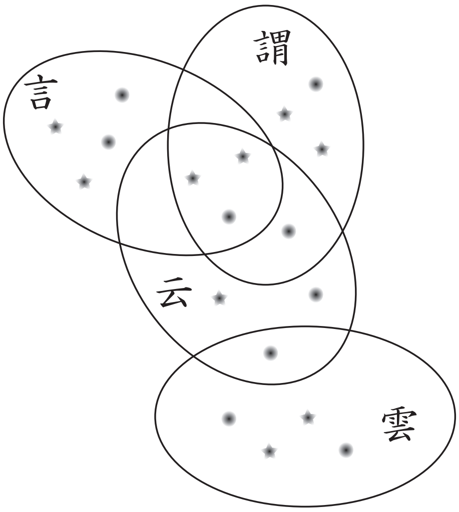


Figure 2: Venn diagram of characters

As we have already explained, a coded character set (CCS) and a code point in the set can also be character features. Those features enable exchanging a character information with the applications that depend on coded character sets. If a character object has only a CCS feature, processing for the character object is the same with processing based on the coded-character model now we are ordinarily using. Namely we can regard the coded-character model as a subset of Chaon model.

3 Overview of CHISE system

Character processing based on Chaon model is to represent each character as a set of character features instead of a code point of a coded character set and process the character by various character features. It indicates that character processing system based on Chaon model is a kind of database system to operate character ontology. So the major targets of CHISE project are (1) character database systems, (2) character database contents and (3) CHISE based applications. CHISE Project is working on the targets and provides some results as free software.

As character database systems (and language bindings), following implementations are available:

libchise is a library to provide fundamental features to operate character database

XEmacs CHISE is a Chaon implementation based on XEmacs [9] (extensible text editing environment with Emacs Lisp interpreter) [Figure 3]

Ruby/CHISE is a Chaon implementation based on Ruby [8] (scripting language)

Perl/CHISE is a Chaon implementation based on Perl (scripting language)

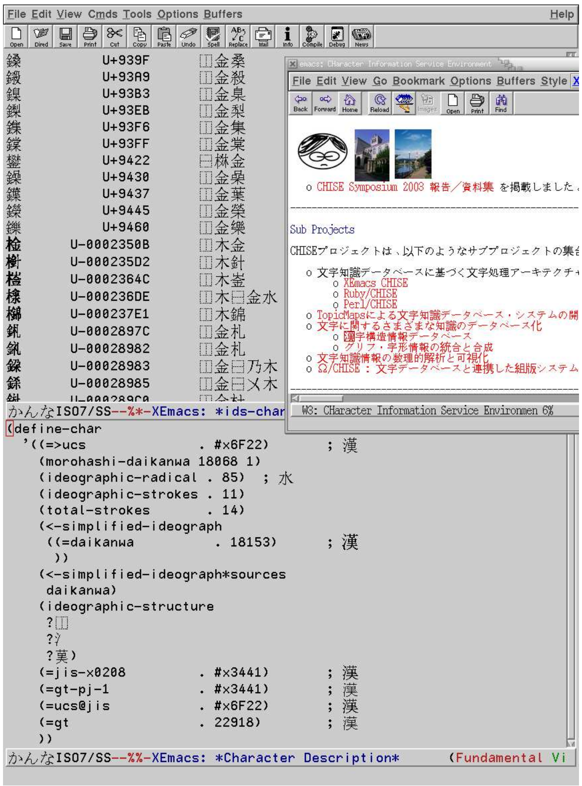
For the Chaon implementations, currently two database contents are available as follows:

CHISE basic character database is a general character database attached to XEmacs CHISE

CHISE-IDS is a database about shapes of Ideographic characters

As CHISE based applications, we need at least editing system, printing system and character database utilities. As editing system, XEmacs CHISE is available. As printing system, the following programs are available:

Ω/CHISE is a multilingual type setting system based on Ω [5] (a multilingual TEX)


File Edit View Cmds Tools Options Buffers
Help
Open Dired Save Print Cut Copy Paste Undo Spell Replace Mail Info Compile Debug News
emacs: CHaracter Information Service Environment
File Edit View Go Bookmark Options Buffers Style X
Back Forward Home Reload Images Open Print Find

```
鎟           U+939F      ⿰金桑
鎩           U+93A9      ⿰金殺
鎳           U+93B3      ⿰金臬
鏫           U+93EB      ⿰金梨
鏶           U+93F6      ⿰金集
鏿           U+93FF      ⿰金棠
鐢           U+9422      ⿱楙金
鐰           U+9430      ⿰金桌
鐷           U+9437      ⿰金葉
鑅           U+9445      ⿰金榮
鑠           U+9460      ⿰金樂
𣔋       U-0002350B      ⿰木金
𣗒       U-000235D2      ⿰木針
𣙌       U-0002364C      ⿰木崟
𣛞       U-000236DE      ⿰木⿱金水
𣟡       U-000237E1      ⿰木錦
𨥼       U-0002897C      ⿰金札
𨦂       U-00028982      ⿰金札
𨦃       U-00028983      ⿰金⿱乃木
𨦅       U-00028985      ⿰金⿱乂木
かんなISO7/SS--%*-XEmacs: *ids-char
```

```
(define-char
  '((=>ucs                    . #x6F22)        ; 漢
    (morohashi-daikanwa 18068 1)
    (ideographic-radical . 85)  ; 水
    (ideographic-strokes . 11)
    (total-strokes       . 14)
    (<-simplified-ideograph
     ((=daikanwa              . 18153)         ; 漢
      ))
    (<-simplified-ideograph*sources
     daikanwa)
    (ideographic-structure
     ?⿰
     ?氵
     ?茣)
    (=jis-x0208             . #x3441)          ; 漢
    (=gt-pj-1               . #x3441)          ; 漢
    (=ucs@jis               . #x6F22)          ; 漢
    (=gt                    . 22918)           ; 漢
    ))
かんなISO7/SS--%%-XEmacs: *Character Description*      (Fundamental Vi
```

```
o CHISE Symposium 2003 報告／資料集 を掲載しました。

Sub Projects

CHISEプロジェクトは、以下のようなサブプロジェクトの集合

  o 文字知識データベースに基づく文字処理アーキテクチャ
      o XEmacs CHISE
      o Ruby/CHISE
      o Perl/CHISE
  o TopicMapsによる文字知識データベース・システムの開
  o 文字に関するさまざまな知識のデータベース化
      o 漢字構造情報データベース
      o グリフ・字形情報の統合と合成
  o 文字知識情報の数理的解析と可視化
  o Ω/CHISE : 文字データベースと連携した組版システム
W3: CHaracter Information Service Environmen 6%
```

Figure 3: XEmacs CHISE

chise2otf is a converter to process CHISE texts with pLaTeX[7] + OTF package [6]

chise-tex.el is like chise2otf, but it is implemented as a coding-system of XEmacs CHISE (written by Emacs Lisp)

As character database utilities, there are some emacs lisp programs, Ruby scripts attached to Ruby/CHISE and Perl scripts attached to Perl/CHISE.

In addition, the following system is available:

Kage is an automatic Ideographic glyph generating system [10]

Currently Kage is used in Ω/CHISE and chise2otf.

4 XEmacs CHISE

XEmacs CHISE (Figure 3), Ruby/CHISE and Perl/CHISE provide operations about character features based on Chaon model. In this paper, I describe a overview of them in the XEmacs CHISE case as an example.

4.1 Character representation

XEmacs CHISE represents a character as a character object. A character object is a first class object, just like a character in XEmacs, and completely a different type from the integer.

For programmers of Emacs Lisp applications, a character is represented by a set of character features. The character feature is a pair of a feature key and its value. The feature key must be a symbol and the value can be any Lisp object, that is, a character object itself can be a feature value of other characters, which makes it easy to represent relation network among characters. Namely a character is represented by an association-list of Lisp. In XEmacs CHISE, such kind of association-list which represents a character by its features is named "character-specification (char-spec)".

Each character also has an unique identifier called “character-id” although it is usually hidden from Emacs Lisp users.

4.2 Character object related functions

In order to define and handle a character and character features, XEmacs CHISE provides the following built-in functions.

Function define-char (*char-spec*)

defines a character object that has a set of character feature *features*, and returns the object. *char-spec* should be an association-list.

[Example]

```
(define-char
  '((name               . "CJK RADICAL MEAT")
    (general-category   symbol other)
    (bidi-category      . "ON")
    (mirrored           . nil)
    (ideographic-radical . 130) ; 肉 (radical)
    (ideographic-strokes . 0) ; (body strokes)
    (total-strokes       . 4) ; (total strokes)
    (<-ideographic-component-forms
     ((=ucs               . #x8089)     ; 肉
      ))
    (=ucs               . #x2EBC)       ; 月
    (->subsumptive ; (included variants)
     ((=gt                . 37857)
      (=gt-pj-6           . #x3879)
      (=daikanwa          . 29237)
      )
     ((=ucs@unicode       . #x2EBC)
      )
     ((=big5-cdp          . #x8A73)
      )
     ((=big5-cdp          . #x8958)
      (=gt-k              . 00417)
      (=gt-pj-k1          . #x377D)
      ))
    ))
```

Function get-char-attribute (*character feature* ***&optional*** *default-value*)

returns a value of a character feature specified by the key *feature* of a character object *character*.

If the value of *feature* is not defined, *default-value* is returned. If *default-value* is omitted, nil is returned.

[Example]

```
(get-char-attribute ?\u2EBC 'name)
→ "CJK RADICAL MEAT"
```

Function put-char-attribute (*character feature value*)

adds or changes the value of a feature of a character. This function sets a Lisp object *value* to a value of the character feature specified by the key *feature* of a character object *character*.

[Example]

```
(get-char-attribute ?あ 'foo)
→ nil
(put-char-attribute ?あ 'foo 1)
→ 1
(get-char attribute ?あ 'foo)
→ 1
```

Function remove-char-attribute (*character feature*) removes character feature *feature* from character object *character*.

Function find-char (*char-spec*)

retrieves the character that has specified features *char-spec*.

XEmacs CHISE provides a map function for character features also. This function aims at finding characters with certain character features or processing characters using its character features.

Function map-char-attribute (*function feature*)

This function maps *function* over entries in *feature* (an association-list). *Function* is called with two arguments, a key and a value in the list repeatedly, until all the pairs in *feature* is used up.

Function char-attribute-alist (*character*)

returns the features of the character *character*. Every feature of a character is retrieved by this function.

Function char-attribute-list ()

returns the list of all existing character features except coded character sets.

XEmacs CHISE has on-memory database per each process besides the CHISE character database shared in the CHISE environment. The on-memory database works as a kind of cache memory for the external database. If a character feature is not found in the on-memory database, the feature value is read from the external database and the value is stored into the on-memory database. If a character feature is found in the on-memory database, XEmacs CHISE does not access the external database.

Modification functions for characters or their features of XEmacs CHISE, such as put-char-attribute, work for on-memory database. However it is volatile. So XEmacs CHISE has a function to save the character data.

Function save-char-attribute-table (*feature*)

saves each character's value of character feature *feature* into the CHISE character database.

XEmacs CHISE has functions to clear character data in the on-memory database to be able to reread from the CHISE database.

Function reset-char-attribute-table (*feature*)

clears character feature *feature* of every character in the on-memory database to be able to reread each value of the *feature* from the CHISE database.

Function reset-charset-mapping-table (*coded-charset*)

clears decoding-table of *coded-charset* in the on-memory database to be able to reread from the CHISE database.

XEmacs CHISE has a function to read every character's value of a specified feature from the CHISE database at a burst.

Function load-char-attribute-table (*feature*)

reads every character's value of a specified *feature* from the CHISE database at a burst.

In addition, there is a function to register a character feature.

Function mount-char-attribute-table (*feature*)

registers character feature name *feature* as a target to read from the CHISE character database.

By the way, Ruby/CHISE and Perl/CHISE don't have on-memory database, so written character data are written into the CHISE database directly. So there are no functions to sync between on-memory database and the CHISE database.

5 Character features

5.1 Categorization

In the character processing based on Chaon model, it is important to analyze characters and their various properties and behaviors and represent them as character features. We can find sundry properties and behaviors of characters, and we can use infinite kind of character features. However common character database requires a guideline about character features. So we think that it is feasible to regard each character feature as an abstraction of an operation for characters.

In the point of view, character features can be categorized like following:

1. general character properties (such as descriptions of dictionaries)
2. mappings for character IDs
3. relations between characters

For example, radicals, strokes and phonetic values can be classed into the category 1, code points of UCS [2] can be classed into the category 2 and relations between character variants can be classed into the category 3.

Information of the category 2 is used for processing about character codes, such as code conversion. Processing about character codes consists of two kind of operations: encoding and decoding. To encode a character by a CCS is to get the CCS feature's value in the character. To decode a code-point of a CCS is to search a character whose value of the CCS feature is the code-point. Processing about character codes should be fast, so the CHISE character database has special indexes for decoding.[1]

For the processing about character variants, information of the category 3 is used.

5.2 Description for complex information

For development of a general purpose character database, we may find some cases that there are different kind of usages, purposes, applications, sources, interpretations, theories, etc. so it is hard to chose one feature value and we want to provide alternative values. In that cases, we may want to add metadata, such as sources of the values. To resolve the problem, we have to introduce structured feature value or structured feature name (key).

To represent structured feature values, a format named "character reference (char-ref)" is used in CHISE. It is a kind of property-list of S-expression (Lisp), property name indicates kind of metadata and property value indicates its data. As a special property name, :char is reserved to indicate a character which is added the metadata. Currently :sources is defined to indicate information source.

CHISE also has a format to represent structured feature names. In the structured feature names, "domain identifiers" and/or "metadata identifiers" are added to ordinary (base) feature names. The format is defined as following definitions:

```
<concrete feature name>
      := <base feature name> @ <domain identifier>
      |   <concrete feature name> / <domain identifier>

<metadata feature name>
      := <concrete feature name> * <metadata identifier>
```

For example, when total strokes is represented by character feature total-strokes and ucs is used as a domain identifier, concrete feature name is total-strokes@ucs. When source is represented by metadata identifier sources, total-strokes@ucs's source is represented by metadata feature name total-strokes@ucs*sources.

If there is a correspondence between different kind of features, such as radical and body-strokes, we can represent the correspondence by a domain identifier. For example, when radical is represented by ideographic-radical and body-strokes is represented by ideographic-strokes, two concrete feature names

ideographic-radical@ucs
ideographic-strokes@ucs

are corresponding.

[1] For the special treatment, we distinguish the category 1 and 2, but it seems that there are no essential differences.

5.3 Inheritance of character definition

If we construct a large scale character database including a lot of character variants, inheritance of character definition is good way to avoid to write a lot of common features. So CHISE introduces four special features to represent parent and child relations:

<–subsumptive defined character is a child of each character indicated by its value

<–denotational likewise

–>subsumptive each character indicated by its value is a child of the defined character

–>denotational likewise

6 Database contents

Character database is a fundamental part of the Chaon character representation model. Users can, of course, freely define or modify characters by adding new character features, but a rich and accurate database would be a great place to start, and it will also attract new users. We have thus developed a standard character database for Chaon implementations. Currently two database distributions are available:

1. CHISE basic character database

2. Database about structure information of Ideographs (CHISE-IDS)

The former is a basic character database attached in XEmacs CHISE which is realized by a collection of define-chars while the later is a database for Ideographs to represent information about shapes which is represented by "Ideographic Description Sequence (IDS)" format defined in ISO/IEC 10646-1:2000 [2].

Structure information of Ideographs is information about combinations of components of Ideographs. A lot of Ideographs can be represented by a combination of components, so the information is a useful. It is not only representation of abstract shapes, but also related with semantic values and/or phonetic values. So we planned to develop a database about structure information of Ideographs for every Ideograph which consists of combination of components. In the 2001 fiscal year, we realized CHISE-IDS database with supporting of "Exploratory Software Project (未踏ソフトウェア創造事業)" run by IPA (Information-technology Promotion Agency, Japan) [11]. Currently it supports CJKV Unified Ideographs and Extension A of ISO/IEC 10646-1:2000 [2] and Extension B of ISO/IEC 10646-2 [3]. We are also working for representative glyph image of JIS X 0208:1990 and Daikanwa Dictionary.

Before we developed CHISE-IDS database, there are some databases including structure information of Ideographs: CDP database [1] by Academia Sinica, database about gaiji (private used character) used in CBETA and "Konjaku Mojikyo" [4]. These databases use original formats so it is not easy to convert to IDS format. Konjaku Mojikyo is a proprietary software so their data are not opened for the public. In the view of licence, CDP database and CBETA database are available with free software licenced under the term of GPL while Konjaku Mojikyo is not. So we converted CDP database and CBETA database to IDS and integrate them with CHISE database.

Currently CHISE basic character database (is a part of XEmacs CHISE) and CHISE-IDS package are distributed separately. However CHISE-IDS package provides an installer to integrate CHISE-IDS database files with CHISE basic character database. CHISE-IDS package also have some utility programs to use structure information of Ideographs in XEmacs CHISE:

ids.el IDS parser

ids-read.el utility to read CHISE-IDS database files into XEmacs CHISE

ids-dump.el utility to dump structure information of Ideographs stored in XEmacs CHISE (represented by character feature ideographic-structure) into CHISE-IDS database format

ids-util.el utility to convert structure information of Ideographs into other representative glyph images corresponding with specified domains

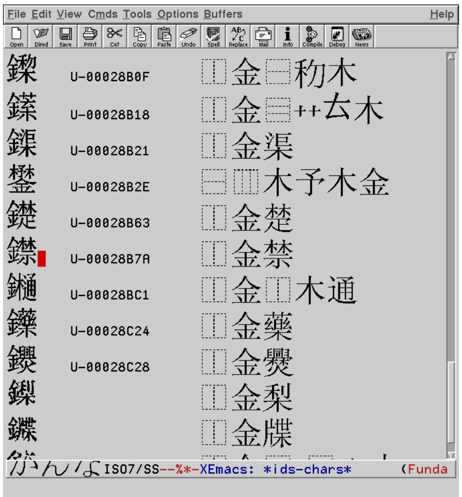
ids-find.el utility to search Ideographs by components

Currently, ids-find.el has two commands to search Ideographs by components named: ids-find-chars-including-components (= ideographic-structure-search-chars) and ids-find-chars-covered-by-components.

If you type

M-x ideographic-structure-search-chars [CR] components [CR]

Ideographs that have every component are displayed. (Figure 4)



```
𨬏    U-00028B0F    ⿰金⿱𥝌木
𨬘    U-00028B18    ⿰金⿳++厷木
𨬡    U-00028B21    ⿰金渠
𨬮    U-00028B2E    ⿱⿲木予木金
𨭣    U-00028B63    ⿰金楚
𨭺    U-00028B7A    ⿰金禁
𨯁    U-00028BC1    ⿰金⿰木通
𨰤    U-00028C24    ⿰金藥
𨰨    U-00028C28    ⿰金爨
[illegible]                  ⿰金梨
[illegible]                  ⿰金牒
```

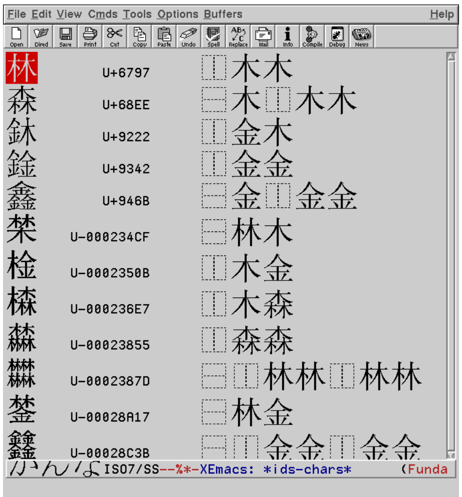
ISO7/SS--%*-XEmacs: *ids-chars*   (Funda

Figure 4: result of M-x ideographic-structure-search-chars [CR] 木金 [CR]

If you type

M-x ids-find-chars-covered-by-components [CR] components [CR]

Ideographs that consists of one or more usage of every component are displayed. (Figure 5).



```
林          U+6797    ⿰木木
森          U+68EE    ⿱木⿰木木
鈢          U+9222    ⿰金木
鍂          U+9342    ⿰金金
鑫          U+946B    ⿱金⿰金金
𣓏    U-000234CF    ⿱林木
𣔋    U-0002350B    ⿰木金
𣛧    U-000236E7    ⿰木森
𣡕    U-00023855    ⿰森森
𣡽    U-0002387D    ⿱⿰林林⿰林林
𨨗    U-00028A17    ⿱林金
𨰻    U-00028C3B    ⿱⿰金金⿰金金
```

ISO7/SS--%*-XEmacs: *ids-chars*   (Funda

Figure 5: result of M-x ids-find-chars-covered-by-components [CR] 木金 [CR]

7 Conclusion

I have described a brand new character processing model Chaon and overview of CHISE project and character processing in CHISE systems such as XEmacs CHISE. Chaon character representation model is powerful and radical enough to solve the problems that the present coded character model has, and the implementation of XEmacs CHISE or other CHISE systems have shown that the model is feasible enough to build a application on.

Chaon model gives users freedom to create, define and exchange characters of their need, as it is easy to change character databases or modify character features dynamically. XEmacs CHISE provides a good framework to experiment the character representation. With the CHISE basic character database, XEmacs CHISE can handle various characters including characters defined in Unicode. Even if a character is not defined in Unicode, users can add it into CHISE database to define it by its features. Users can handle each character based on their point of view or policy. For example, XEmacs CHISE provides some unification rules or mapping policies about Unicode. With the CHISE-IDS database, users can search Ideographs easily. This method is also available for non-Unicode characters.

Currently, CHISE project provides basic elements to process Chaon based text: text editor (XEmacs CHISE), scripting languages (Ruby/CHISE and Perl/CHISE) and type setting system (Ω/CHISE). You can try KNOPPIX/CHISE which is a DVD bootable GNU/Linux system including XEmacs CHISE and Ω/CHISE. Its image is available at `http://kanji.zinbun.kyoto-u.ac.jp/projects/chise/dist/KNOPPIX/`.

CHISE project is an open source project, so its results are distributed as free software. Information about CHISE project is available at:

- `http://cvs.m17n.org/chise/`
- `http://kanji.zinbun.kyoto-u.ac.jp/projects/chise/`

These WWW pages, various programs and data are managed by CVS (a kind of revision control system), so users can get the latest snapshot. There are mailing-lists about CHISE project: for English and Japanese. If you are interested in CHISE project, please join to the lists.

References

[1] 漢字庫. `http://www.sinica.edu.tw/~cdp/zip/hanzi/hanzicd.zip`.

[2] International Organization for Standardization (ISO). *Information technology — Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane (BMP)*, March 2000. ISO/IEC 10646-1:2000.

[3] International Organization for Standardization (ISO). *Information technology — Universal Multiple-Octet Coded Character Set (UCS) – Part 2: Supplementary Planes*, November 2001. ISO/IEC 10646-2:2001.

[4] 今昔文字鏡. http://www.mojikyo.com/.

[5] The Omega typesetting and document processing system. http://omega.cse.unsw.edu.au:8080/.

[6] Open Type font 用 VF. http://psitau.at.infoseek.co.jp/otf.html.

[7] Ascii 日本語 TEX(pTEX). http://www.ascii.co.jp/pb/ptex/.

[8] The object-oriented scripting language Ruby. http://www.ruby-lang.org/.

[9] XEmacs. http://www.xemacs.org/.

[10] 上地 宏一. 漢字フォント自動生成サーバ “影 KAGE” の構築 — 文字コードの枠組みを越える次世代漢字処理の提案 —. **漢字文献情報処理研究**, 3:143–147, 2002.

[11] 守岡 知彦 and クリスティアン・ウィッテルン. 文字データベースに基づく文字オブジェクト技術の構築. In **情報処理振興事業協会 平成 *13* 年度 成果報告集**. 情報処理振興事業協会, 2002. http://www.ipa.go.jp/NBP/13nendo/reports/explorat/charadb/charadb.pdf.

基于字符集的中文信息处理

翟喜奎
（国家图书馆 数字图书馆管理处）

文摘：根据在古籍文献处理时所遇到的中文信息处理问题提出了基于字符集的中文信息处理。基于字符集的中文信息处理应包括汉字排序，汉字可有多种排序标准(拼音、部首、笔画、四角号码等等)；规范检索，用简体、繁体、异体字都能统一检索；汉字输入方法；支持大字符集的显示等等。探讨基于字符集的中文信息处理方法。

关键词： 中文信息处理 汉字字符集 汉字排序 汉字属性标准

一、中文字符集的发展

1. GB 2312-80《信息交换用汉字编码字符集－基本集》

中国于1980 年3 月颁布了第一个汉字编码字符集标准，即GB 2312-80《信息交换用汉字编码字符集－基本集》。该标准符合ISO 2022编码体系结构。1981 年5 月1 日开始正式实施，它奠定了中国中文信息处理技术的发展基础。

随着GB 2312-80的颁布，中国颁布了相应的《15×16、24×24、32×32汉字点阵字模集及字模数据集》。所谓点阵字形，就是以点的形式来表现的字符或汉字的形态。15×16点阵字形，可以表示《信息交换用汉字编码字符集－基本集》中的绝大部分汉字。由于15×16的点阵字形只能表示横向笔画和竖向笔画都不超过八笔的汉字。如果一个汉字的横向笔画或者竖向笔画超过了八笔（如“量、酬”等字），在点阵字模就容纳不下。在《基本集》中，这样的汉字共有138个，只好压缩笔画做变通处理。15×16点阵字形适于屏幕显示，作校对之用。

24×24点阵字形，可以全部表示《基本集》中的6763个汉字的笔形结构，用不着压缩笔画，而且能够保持横细竖粗的宋体风格，适用于针式打印、喷墨打印，是一种很有使用价值的点阵字形。

32×32点阵字形比24×24点阵字形更能体现宋体风格，能完整地表现汉字的笔锋，使撇笔和捺笔自然婉转，舒畅流利，字体质量较高。

1992年中国颁布了矢量汉字的字模集及数据集：

- GB/T 13844-1992图形信息交换用矢量汉字 单线宋体字模集及数据集
- GB/T 13845-1992图形信息交换用矢量汉字 宋体字模集及数据集
- GB/T 13846-1992图形信息交换用矢量汉字 仿宋体字模集及数据集
- GB/T 13847-1992图形信息交换用矢量汉字 楷体字模集及数据集
- GB/T 13848-1992图形信息交换用矢量汉字 黑体字模集及数据集

1992年、1993年中国颁布了高精度点阵汉字标准：

- GB/T 14242-1993 信息交换用汉字64×64点阵黑体字模集及数据集
- GB/T 14243-1993 信息交换用汉字64×64点阵楷体字模集及数据集
- GB/T 14244-1993 信息交换用汉字64×64点阵仿宋体字模集及数据集

- GB/T 14245-1993 信息交换用汉字64×64点阵宋体字模集及数据集
- GB/T 14717-1993 信息交换用汉字128×128点阵宋体字模集及数据集
- GB/T 14718-1993 信息交换用汉字128×128点阵黑体字模集及数据集
- GB/T 13443-1992 信息交换用汉字128×128点阵楷体字模集及数据集
- GB/T 13444-1992 信息交换用汉字128×128点阵仿宋体字模集及数据
- GB/T 14719-1993 信息交换用汉字256×256点阵宋体字模集及数据集
- GB/T 14720-1993 信息交换用汉字256×256点阵黑体字模集及数据集
- GB/T 13445-1992 信息交换用汉字256×256点阵楷体字模集及数据集
- GB/T 13446-1992 信息交换用汉字256×256点阵仿宋体字模集及数据集

2. 编码字符集的繁体字和简体字对应编码

1984 年“全国计算机与信息处理标准化技术委员会”提出编码字符集的繁体字和简体字对应编码的原则，并做出了制定六个信息交换用汉字编码字符集的计划。这六个集分别命名为基本集（GB2312-80）、第一辅助集(辅一)、第二辅助集(辅二)、第三辅助集(辅三)、第四辅助集(辅四)、第五辅助集(辅五)。其中，基本集、辅二集、辅四集是简体字集，辅一集、辅三集、辅五集分别是基本集、辅二集、辅四集的繁体字映射集，且简/繁字在两个字符集中同码(个别简/繁关系为一对多的汉字除外)。这六个集均采用双七位编码方式，但为了避开ASCII表中的控制码，每个七位只选取了94 个编码位置。所以每张代码表分94 个区和94 个位。其中前15 区作为拼音文字及符号区或保留未用，16 区到94 区为汉字区。第一辅助集(辅一)、第二辅助集(辅二)、第三辅助集(辅三)、第四辅助集(辅四)、第五辅助集(辅五)分别于1987年至1991年颁布。

- GB 12345-90《信息交换用汉字编码字符集——第一辅助集》
- GB 7589-87《信息交换用汉字编码字符集——第二辅助集》
- GB13131-1991《信息交换用汉字编码字符集——第三辅助集》
- GB 7590-87《信息交换用汉字编码字符集——第四辅助集》
- GB13132-1991《信息交换用汉字编码字符集——第五辅助集》

3. GB13000《信息技术通用多八位编码字符集》

1990年中国颁布了GB13000《信息技术通用多八位编码字符集》。

随着国际间的交流与合作的扩大，信息处理应用对字符集提出了多文种、大字量、多用途的要求。1993年国际标准化组织发布了ISO/IEC 10646-1《信息技术通用多八位编码字符集第一部分体系结构与基本多文种平面》。我国等同采用此标准制定了GB 13000.1-1993。该标准采用了全新的多文种编码体系，收录了中、日、韩20902个汉字，是编码体系未来发展方向。

4. GBK 编码字符集

1995 年12 月完成GBK 规范。GBK 编码是GB2312-80 国标码的扩充（其中GB 表示国标，K 表示扩展）。但是GBK编码本身不是国家标准。该编码规范完全兼容GB2312-80。

5. GB18030-2000《信息交换用汉字编码字符集基本集的扩充》

2000年3月中国颁布了国家标准GB18030-2000《信息交换用汉字编码字符集基本集的扩充》是我国继GB2312-1980和GB13000-1993之后最重要的汉字编码标准，是未来我国计算机系统必须遵循的基础性标准之一，该标准是国家强制性标准。在中国大部分计算机系统仍然采用GB 2312编码。GB 18030与GB 2312一脉相承，较好地解决了旧系统向新系统的转换问题，并且改造成本较小。从我国信息技术和信息产业发展的角度出发，考虑到解决我国用户的需要及解决现有系统的兼容性和对多种操作系统的支持，采用GB 18030是我国目前较好的选择，而GB 13000.1更适用于未来国际间的信息交换。考虑到GB 18030和GB 13000的兼容问题，标准起草组编制了GB 18030与GB 13000.1的代码映射表，使得两个编码体系可以自由转换。同时，还开发了GB 18030基本点阵字型库。

6. 台湾字符集

《中文资讯交换码CCCII》。CCCII是Chinese Character Code for Information Interchange的缩写，是经台湾中研院中美会及国科会等单位支持,于1979年12月集合由台湾图书馆学者、文字学家及电脑专家组成“国字整理小组”提出的汉字编码。经过一些修改，被美国采纳为美国国家标准ANSI Z39.64-1989，同时也被称为东亚字符编码（EACC）用于图书馆书籍目录方面。

《通用汉字标准交换码CNS 11643》。1983 年 10 月，台湾科学委员会、教育部国语推行委员会、中央标准局、行政院主计处电子资料处理中心共同制定了《通用汉字标准交换码》（Chinese Ideographic Standard Code for Information Interchange，简称 CISCII码），经试用修订，1986年 8月 4日由台湾中央标准局公布为法定标准，标准编号为 CNS 11643。这一标准于 1992年 5月 21日重新修订公布，更名为《中文标准交换码》（Chinese Standard Interchange Code）。1995年 1月 4日，台湾中央标准局又公布了 CNS 11643-1《中文标准交换码使用方法》。

上述两个汉字字符集，CNS 11643为通用的标准交换码，适用面较广。CCCII使用面相对较窄，相当于行业规范。

BIG-5 码是 1984 年台湾资讯工业策进会根据《通用汉字标准交换码》制订的编码方案。

7. 日本字符集标准

1978年，日本政府公布了日本工业标准JIS C 6226-1978《信息交换用汉字字符集》。该标准于1983年进行了修订，新增加了4个汉字，并将该标准编号改为JIS X 0208-1983。

1990年日本发布了第二个日本汉字编码字符集标准JIS X 0212-1990，作为日本汉字交换码辅助集。

1993年日本发布了第三个日本汉字编码字符集标准JIS X 0221-1993，该标准是有20,902个汉字的编码标准。接着，有了1996年的《表外汉字字体表试案》，日本政府在公布该方案时，并且声明，它“是法令、公用文书、报纸、杂志、广播电视等一般社会生活中，使用表外汉字字体的依据”，“这个字体表将明治以来传统的印刷文字字体（并非《康熙字典》字体的本身，而是以《康熙字典》为依据作成的明治以来的铅字字体即《康熙字典体》）置于印刷标准字体的地位。”

目前，日本最新汉字编码字符集标准是JIS X 0221-1:2001。

8. 韩国字符集标准

1987年韩国制定了韩国标准编码字符集KS C 5601-1987，共有8224个字符。1991年韩国制定了编码字符集的辅助集KS C 5657-1991，增收汉字2856个。

9．ISO/IEC 10646与UNICODE

国际标准化组织（ISO）于1984年开始研究、制定《信息技术 通用多八位编码字符集（UCS）》国际标准，即ISO/IEC 10646。1993年5月，该标准的第一部分：体系结构与基本多文种平面（即ISO/IEC 10646.1）正式发布。前后经历了九年的时间。ISO/IEC 10646-1的第二版更加全面，即ISO/IEC 10646-1：2000。与ISO/IEC 10646-1：2000 等同的工业标准是Unicode3.0，Unicode 是Universal Code 的简称，即统一编码。除了作为ISO/IEC10646 编码的一种称谓外，Unicode 同时还是由HP，IBM，APPLE，MICROSOFT 等一些国际知名企业组成的一个联盟的名称。该联盟的主要宗旨就是要通过市场手段推进多文种的统一编码，因此称为Unicode。它的广泛使用将会使得软件开发费用大幅度降低，开发更为快捷，可共享资源更为丰富，使用者的投入也将大幅度降低，便于推广。软件可以实现一个版本的世界范围内通用，不再需要多个版本、多种语言的产品了。目前兼容和支持该标准的已有许多大型厂商，如微软、苹果、SUN、甲骨文等国际性软件公司。

目前，ISO/IEC 10646国际标准的最新版本是2003年修订的ISO/IEC 10646:2003等同的工业标准是Unicode 4.0。

二、中文字符集的基本内容

1．GB 2312-80《信息交换用汉字字符集—基本集》

GB2312-80《信息交换用汉字字符集—基本集》 收录汉字信息交换用的基本图形字符，采用一字一码的原则，实现简化汉字6763 个，总计7445 个图形字符。具体包括：一般字符202 个，序号60 个，数字22 个，英文字母大小写共52 个，日文平假名169 个，希腊字母大小写共48 个，俄文字母大小写共66 个，汉语注音符号37 个，一二级汉字共6763 个。其中：一级常用汉字3 755 个，按照汉语音排序；二级非常用汉字3008 个，按照偏旁部首排序。该标准的制定和应用为规范、推动中文信息化进程起了很大作用。

■ 双字节编码

■ 范围：A1A1～FEFE

■ A1-A9：符号区，包含682 个符号

■ B0-F7：汉字区，包含6763 个汉字。

2．《汉字国标扩展规范GBK》

《汉字国标扩展规范GBK 》，在MS Windows 9x/Me/NT/2000、IBM OS/2 的系统中广泛应用。是GB2312 国标码的扩充。它是国家技术监督局1995 年为中文Windows 95所制定的新的汉字内码规范(其中GB 表示国标,K 表示扩展)。该规范在字汇一级上支持ISO10646和GB13000 中的全部中日韩（CJK）汉字，并与国家标准GB2312-80 信息处理交换码相兼容。

■ 双字节编码，GB2312-80 的扩充，在码位上和GB2312-80 兼容

■ 范围：8140～FEFE（剔除xx7F）共23940 个码位

■ 包含21003 个汉字，包含了ISO/IEC 10646-1 中的全部中日韩汉字。

3．GB 12345-90《信息交换用汉字编码字符集—第一辅助集》

国家标准GB1234-90《信息交换用汉字编码字符集—第一辅助集》于1990 年发布，是与基本集对应的繁体字字符集，共收图形字符7583个，其中前15区除收集了GB 2312-80中前15 区内收的全部字符外，又增收了35 个竖排标点符号和汉语拼音符号共717个字符。从

16 区至91 区共收6866 个繁体汉字。目的在于规范必须使用繁体字的各种场合，以及古籍整理等。一级汉字数和二级汉字数都与GB2312-80 相同，另有103 个繁体字是属于简/繁为一对多的字（比GB2312 多103 个字，其它厂商的字库大多不包括这些字）。对于简/繁一对多的情况，则选一个最通用的繁体字码置于与基本集中该字相对应的码位，其余的则按拼音序编码于88和89 区。实际上可以将GB1234-90理解为GB2312-80的繁体字对应版。如果在同一套计算机系统中要支持简、繁体字共存，只能采取两个分别代表GB2312-80 和GB12345-90 的代码页。

4．GB 7589-87《信息交换用汉字编码字符集—第二辅助集》

GB 7589-87《信息交换用汉字编码字符集—第二辅助集》是作为基本集的补充而编制的，收入通用规范的简体汉字，收字7237 个，以201个部首为序排列，部首次序按笔画数排列，同部首字按部首以外的笔画数排列，同笔画数的字以笔形顺序(横、直、撇、点、折)为序。不收繁体字和被淘汰异体字。

5．GB 13131-1991《信息交换用汉字编码字符集—第三辅助集》

辅三集是辅二集对应的繁体字字符集，简繁体有一一对应关系，收字与辅二集相同。

6．GB 7590-87《信息交换用汉字编码字符集—第四辅助集》

辅四集是作为基本集的补充而编制的，均收通用规范的简体汉字，收字7039 个，都以201个部首为序排列，部首次序按笔画数排列，同部首字按部首以外的笔画数排列，同笔画数的字以笔形顺序(横、直、撇、点、折)为序。不收繁体字和被淘汰异体字，第二辅助集和第四辅助集共约有4200 多个字是经过类推简化得到的，提高了整个字符集的规范性，但降低了字符集的实用性。

7．GB 13132-1991《信息交换用汉字编码字符集—第五辅助集》

辅五集是辅四集对应的繁体字字符集，简繁体有一一对应关系，收字与辅四集相同。

8．GB 13000《信息技术通用多八位编码字符集》（ISO/IEC10646）

GB13000是ISO/IEC10646的等同标准。国际标准化组织为了将世界各民族的文字进行统一编码，制定了UCS 标准。根据这一标准，中、日、韩三国共同制定了《CJK 统一汉字编码字符集》，其国际标准号为：ISO/IEC10646，我国国家标准号为：GB13000。该汉字编码字符集就是通常人们所说的大字符集，它编入了20902 个汉字，收集了大陆一、二级字库中的简体字，台湾《通用汉字标准交换码》中的繁体字，58 个香港特别用字和92 个延边地区朝鲜族“吏读”字，甚至涵盖了日文与韩文中的通用汉字，满足了方方面面的需要。

9．GB 18030-2000《信息技术信息交换用汉字编码字符集基本集的扩充》

GB18030标准是中国政府于2000年3月颁布的最新中文汉字编码标准，是我国继GB2312-1980 和GB13000-1993 之后最重要的汉字编码标准，是未来我国计算机系统必须遵循的基础性标准之一。GB18030 收录了27484 个汉字。双字节部分收录内容主要包括GB13000.1 全部CJK 汉字20902 个、有关标点符号、表意文字描述符13 个、增补的汉字和部首/构件80 个、双字节编码的欧元符号等。四字节部分收录了上述双字节字符之外的，包括CJK 统一汉字扩充A 在内的GB 13000.1 中的全部字符。GB18030 编码空间约为160 万码位，目前已编码的字符约2.6 万。随着我国汉字整理和编码研究工作的不断深入，以及国际标准ISO/IEC 10646 的不断发展，GB18030 所收录的字符将在新版本中增加。

10. 统一码（Unicode）

统一码(Unicode)与ISO 10646 国际编码标准互相兼容。统一码是由一个名为Unicode学术学会的机构制订的字符编码系统。该系统是为了将世界上几十种紊乱的字符编码整合在一起，以期减少各电脑商开发国外市场时遇到的问题，美国各大电脑厂商组成了策进会，以推广一个世界通行的编码体制，以支持世界主要语文的书面文本的交换、处理及显示。Unicode学术学会的成员大部分为计算机软硬件的供货商。

在1991 年，国际标准化组织与Unicode 学术学会决定共同制订一套适用于多种语文文本的通用编码标准。自此以后，该两个组织便一直紧密合作，同步发展ISO 10646 国际编码标准及统一码。国际标准化组织提供ISO 10646 国际编码标准内的字符及编码资料，Unicode 学术学会则对这些字符及编码资料提出应用的方法以及语义资料作补充。ISO 10646 国际编码标准与统一码所包含的字符及使用的编码是相同的。统一码可被视为是ISO 10646 国际编码标准的实践版。因此，支持统一码的产品，亦支持ISO 10646 国际编码标准。由Unicode 学术学会制订的统一码3.0 版本，于2000 年2月正式推出。这个版本收纳了49,194 个来自世界各地不同语文的字符，其中包括27,484 个东亚的表意文字(汉字，汉字是经过CJK 整合的，即将中日韩文中相近的汉字用单一的编码，称为统一汉字Unihan，共2 万多个，但并不包含一些罕见的字，如康熙字典中的一些古字)。Unicode 编码有多种实现，常见的有UTF8，UTF16，UCS-2，UCS-4 等，统一码3.0 版本是与ISO/IEC 10646-1:2000 对应的版本。统一码3.1 版本于2001 年3 月推出。这个版本的主要特点是增加了44，946 个新字符，其中42，711 个为表意文字。连同统一码3.0 版本原有的字符，统一码3.1 版本共收录了94，140 个字符，其中表意文字总数超过70，000 个。统一码于2002 年推出的3.2 版本。虽然这个版本包括了1，016 个新字符，但其包含的表意文字则与统一码3.1 版本相同。统一码2003年推出最新4.0版本，统一码4.0版本与ISO 10646:2003 国际编码标准的现行版本完全对应，目前，在网络、Windows 系统和很多大型软件中得到应用。

三、基于字符集的中文信息处理

1. 问题的提出

数字图书馆，是面向未来互联网发展的信息管理模式。以数字资源的制作、存储、管理、传输和服务为主要特征的数字图书馆技术，是21 世纪国际科技文化竞争的焦点之一。数字图书馆涵盖多个分布式、超大规模、可互操作的异构多媒体资源库群，面向社会公众提供全方位的知识服务。可以说，数字图书馆将实现对人类知识的普遍存取，并最终消除人们在信息获取方面的不平等。它既是知识网络，又是知识中心，同时也是一套完整的知识定位系统。

国家数字图书馆工程将在国家图书馆二期工程内建设国家数字图书馆国家中心，并通过应用系统开发实现数字资源采集、加工、处理、存储、归档、组织、发布和利用全过程。

地方志是我国所特有的一种文献形式，其中有6, 300余种、120, 000余册。建国前的旧方志是国家图书馆独具特色的馆藏之一，所存文献数量与品质极高。旧方志数字化是一项重要的人文学术研究基础工程，利用计算机及网络技术进行深入的整理、开发，在当今数字化时代势在必行，它将大幅度地提高大众学习、认识中国古代地方文化的效率，即可以将学者的时间和精力从艰苦而繁琐的爬梳、翻检工作中解放出来，又可以向普通读者打开发掘、发现地方志宝藏的大门。地方志文献的数字化是全部中文文献数字化事业的一个复杂特例，是数字化图书馆事业的一个重要部分。

2004 年，国家图书馆将50 万筒子页地方志文献进行文本化数字处理和版式还原。目的

是要实现地方志文献全文检索。共处理汉字约2.3亿，遇到UNICODE4.0之外的字符集集外字大约4500字。因此，在处理古籍文献时，所使用的汉字数量是很大的。除此之外，规范检索问题即用简体、繁体、异体字都能统一检索；检索到的结果进行汉字排序问题；大字符集汉字输入问题；大字符集汉字显示问题等等都要进行处理。这些都是中文信息处理中所遇到的实际问题。

基于字符集的中文信息处理应包括汉字排序，汉字可有多种排序标准(拼音、部首、笔画、四角号码等等)；规范检索，用简体、繁体、异体字都能统一检索；汉字输入方法；支持大字符集的显示等等。

2．进行汉字属性标准研究

为了解决基于字符集的中文信息处理即汉字排序、规范检索、汉字输入、汉字显示等问题，就要对汉字属性标准进行研究。目前，基于GB13000.1《信息技术通用多八位编码字符集》即(ISO/IEC10646.1-1993）、UNICODE1.0的汉字属性标准研究已经完成，解决的汉字数量是基本集20902个汉字。但是，基于UNICODE4.0（ISO/IEC10646：2003）的汉字属性标准研究，当前还是空白，要解决的汉字数量是扩充A集6582个汉字、扩充B集42711个汉字。要加速该方面的研究，满足数字图书馆资源建设以及实际应用工作的需求。

汉字属性标准研究的基本内容是汉字字型标准化、汉字标准发音、字型特征（包括汉字总笔画数量、汉字起笔至末笔笔形值、部首笔画数量、部首序号、部首外起笔至末笔笔形值、异体字数量、异体字字型等）、各种编码（包括四角号码、输入编码、其他汉字字符集编码等）以及构词和使用频度等。

3．汉字排序

汉字的排序方法是依据我国多年延续下来的传统规范和若干限定条件对汉字进行排序处理。即要使汉字象拉丁字符一样成为有序的集合，在计算机内能够进行比较、计算。我国目前使用的汉字排序方法主要有四种：部首法、汉语拼音法、笔画法和四角号码法。这些方法的规则如下：

部首法是以部首归并汉字的一种排检方法。它是先把汉字按其所属的部首归并集中。部首按笔画数多少排列先后顺序， 笔画数目相同的部首，依起笔笔形(横、竖、撇、点、折)排列先后顺序。同属一个部首的字，其先后顺序仍然是先按部首之外的笔画数排列， 部首之外的笔画数目相同的，再依起笔笔形顺序排列。

汉语拼音法是按照汉字发音和声调来归并排列汉字的一种方法。它的一般形式是：先按汉字的发音和声调来归并汉字，按字母的序列排序。音、调相同依笔画数多少排列。笔画数相同，再依起笔笔形(横、竖、撇、点、折)排列先后顺序。

笔画法是按照笔画数目及起笔笔形来归并排列汉字的一种方法。它的一般形式是：先按笔画数多少来归并汉字，笔画数相同，再依起笔笔形(横、竖、撇、点、折)排列先后顺序。

四角号码法是一种以数码来代表汉字四角的笔形并据此来排列汉字先后次序的方法。先按四角号码数多少来归并汉字。四角号码相同，依字中“横”笔的多少排列。“横”笔相同，依整体字的笔数排列。整体字的笔数相同，再依起笔笔形(横、竖、撇、点、折)排列先后顺序。

用计算机处理汉字排序问题的规则，见下表：

	因素1	因素2	因素3	因素4	因素5
部首法	部首序号	部首外汉字笔数	部首外汉字起笔至末笔笔形值	内码	
汉语拼音法	汉语拼音	声调	总笔画数	汉字起笔至末笔笔形值	内码
笔画法	总笔画数	汉字起笔至末笔笔形值	内码		
四角号码法	四角号码	横笔数	总笔画数	汉字起笔至末笔笔形值	内码

4. 规范检索

要解决标准正形汉字与繁体字与异体字相互连接问题。建立相互参见对照表，解决规范检索问题。

5. 输入方法

要选择一种或几种适合古籍大字符集的输入方法，解决汉字输入问题。

6. 汉字显示

古籍资源的应用是全球化的问题，需要解决古籍大字符集的显示问题。虽然，系统都支持UNICODE，但是没有扩A、扩B大字符集字库的支持也不能正确显示汉字。因此，需要解决支持大字符集的汉字字库问题。

总之，基于字符集的中文信息处理是当今数字图书馆古籍资源建设和应用的基础。呼吁各界重视基于字符集的中文信息处理的基础研究，再创我国中文信息处理领域的辉煌。

参考文献：

《中文信息处理技术的现状与进展》 通用中文代码国际联合会 1991年3月 Version 2.0

《试论中文文献的有序化》 翟喜奎《现代图书情报技术》1990年 第1期

A Model for Scholarly Collaboration in the Development of On-line Reference Works: The Digital Dictionary of Buddhism

Charles Muller
Toyo Gakuen University
Kyoto University Institute of Humanities Presentation
Beijing January 22, 2004

I. Technical Review

I began the compilation of the Digital Dictionary of Buddhism (DDB) and the CJKV-English Dictionary soon after my entry into graduate school in Buddhist Studies, upon my coming to awareness of the dearth of adequate lexicographical and other reference works in English language for the textual scholar of East Asian Buddhism in particular, and East Asian philosophy and religion in general. I decided, during my first Buddhist and Confucian/Daoist texts readings courses to save everything I looked up, and have continued that practice down to the present, through the course of studying scores of classical texts.

At the time that I began this process, I could not have dreamt of such a thing as the Internet, or even thought of the possibility of having this material available as a digital database—I was simply envisioning the eventual publication of a new, comprehensive printed work. But as developments in the IT world progressed, the newly appearing potentialities gradually began to dawn on me. Then, in 1995, I tasted the Internet, and once I figured out how to insert <*html*> tags at the beginning and end of a text file, I was on my way to preparing these dictionaries for web publication—the first version of which I uploaded in the summer of 1995. Soon after this, the dictionary was discovered on by

Christian Wittern, who promptly downloaded all the files, and applied a basic SGML structure, which is the ancestor of the XML markup system used today.

Due to the lack of widespread popular implementation of SGML, I did not make any special effort to develop this format for a few years. But after 2000, the popularity of XML began to suddenly increase, and so I began to take this format seriously. A major technical turning point in the history of the DDB came in January 2001, when I was contacted by Michael Beddow, a scholar of German Literature who was also an extremely accomplished XML programmer, and who had been using XML for some time already to develop his own only lexicographical project, the Anglo-Norman Dictionary (*http://www.anglo-norman.net*). Michael generated, based on the markup structure of the DDB, an array of indexes that used Xpointers to call up single-entry data units out of large files, each of which contained hundreds of entries. Michael also developed a CJK-Utf-8 search engine.[1]

II. Content Development

In my first presentation of the DDB at the meeting of the Electronic Buddhist Text Initiative (EBTI) in 1996, the dictionary had 3,200 entries. Today, less than nine years later, the DDB now boasts 35,000 entries, making it by far the largest compilation of its type in the English language, and even larger than some of the best-known Japanese works, such as Oda's *Bukkyō daijiten*. An instrumental factor in this rate of growth is the aid received through JSPS research grants, which allowed us to hire graduate students to help digitize large amounts of data for input. But this stage ended in 2002, and we have entered a new phase, where we are finally receiving large contributions of data from

[1] I have focused here on developments in the DDB, but please note that all of the same technological enhancements have been applied to the CJKV-E.

unselfish collaborators who understand the spirit of the project and its limitless potential for the future. Two of the largest recent individual contributions have come from Prof. KARASHIMA Seishi, who has contributed over 7,000 entries from his research on the *Lotus Sutra*, and from Dr. Stephen Hodge, who has contributed over 2,500 hundred terms from his translation work on the *Yogācārabhūmi-śāstra.* In addition to these unusually large contributions, we have recently been benefiting from a continuous stream of smaller contributions, amendments, and corrections, from an ever-increasing number of scholars.[2]

While the DDB can certainly be viewed as a fairly successful model of the possibilities of online collaboration, it should be made clear that until we set up a mechanism to strongly encourage (perhaps "force" is the better term here) contribution, voluntary data submissions were few and far between. Initially we set up our password access/quota system to deal with hacking and data-theft problems. But we also discovered that we could take advantage of this same system to encourage contributions. Through this system, users who log onto the DDB web site to search for terms are able to freely look up ten items in a 24-hour period. After this, they are greeted by a message telling them that their quota is finished, but that they may gain an unlimited quota password by making a small data contribution.

In earlier days, when the content coverage of the DDB was still rather limited, this strategy did not generate that much response. But during the past year, with the expansion of the coverage to its present number, usage of the resource has also increased. The DDB has become a standard lookup tool for many Buddhist studies specialists—especially those who are doing intensive translation work. It is also used extensively in university classes in North America, and is a basic research tool listed on the syllabi of Buddhist

[2] For a full list of contributors, see *http://www.acmuller.net/credits/credits-ddb.htm*.

Studies courses in such prestigious institutions as Harvard, Stanford, Princeton, Columbia, Berkeley, and other universities. As the DDB grows in both usefulness and in reputation as an essential reference tool for Buddhist studies research, scholars and students are increasingly coming to depend upon it, and thus eventually come to need unlimited access. Most serious scholars already have a large amount of specialized information on their hard drives that can easily be modified to become a DDB entry. All they need, it seems, is a small reason to make this effort, along with a little prodding.

For interested persons who do not have the specialized training to write or edit DDB entries, paid subscriptions are available. This approach was settled upon not with the expectation of making a lot of money, but simply to provide a recourse for persons who demanded full access in one way or another. As a by-product of this offering however, we decided to offer institutional subscriptions as well, and recently a number of major universities have decided to have their libraries subscribe to the DDB, including Columbia, Berkeley, Santa Barbara, and UCLA. While we are happy to gain a small amount of money to put back into the project, at this point, the greatest value of these subscriptions is in the recognition being accorded to the DDB as a primary reference tool. It is especially significant that this reference tool has been put together and produced, not by a major publishing company, but by a group of like-minded scholars.

III. The Structure of a DDB Entry

As mentioned earlier, the DDB uses XML as its basic structural format. The DTD is based loosely on the recommendations of the Text Encoding Initiative, using many of the entities and attributes that are used for lexicons.[3] An entry is divided into three major

[3] The reason that the DDB is not based more fully on TEI is simply that most of the structure was developed before I adequately understood the TEI model. I have thought from time to time about redoing the whole structure according to the TEI DTD, but the retooling of the stylesheets, as well as

sections: (1) A Pronunciation Section, wherein the readings of a Chinese term are provided in various East Asian languages and their romanization systems. (2) A Sense Section, which provides the translation of the term and other explanatory material, and (3) An External References Section, which provides references to the term in a variety of Buddhist Studies reference works. Each of these larger nodes has children nodes, and various other entities contained within. When a user selects a term either via hyperlink or by search engine lookup, and HTML page is generated. One sample page is given below:

numerous other aspects of the web implementation of the data set are simply too daunting for me to seriously consider at this point in time.

言說

[Pronunciations]

[py] yánshuō
[wg] yen-shuo
[hg] 언설
[mc] eonseol
[mr] ŏnsŏl
[kk] ゴンゼツ
[hb] gonzetsu

Meanings

[Basic Meaning:] verbal expression [s.hodge]

Senses:

- (Skt. *vyavahāra*; Tib. *tha snyad*) [s.hodge]
- expresses, recounts; (Skt. *abhi-lap**; Tib. *brjod par 'gyur ba*) [s.hodge]
- expressing, an expression, an utterance; (Skt. *abhilāpa*; Tib. *brjod pa*) [s.hodge]
- discourse; (Skt. *kathā*; Tib. *gtam*) [s.hodge]
- a figurative designation; (Skt. *upacāra*; Tib. *nye bar 'dogs pa*) [s.hodge]
- Language, speech (*vāc*), which is one of the three kinds of permeation of the store consciousness taught in the *Mahāyāna-saṃgraha*. 〔攝大乘論 (T 1593.31.117c3) 〕 [cmuller]
- The usage of language to teach the dharma (*deśanā*). [cmuller]
- Language as synergistic with the mental realm of phenomenal differentiation (*abhilāpa*). [cmuller]

[Dictionary References]

Iwanami Bukkyō jiten 239, 293
Bukkyōgo daijiten (Nakamura)429b
Ding Fubao
Buddhist Chinese-Sanskrit Dictionary (Hirakawa)1072
Bukkyō daijiten (Mochizuki)(v.9-10)1043b
Bukkyō daijiten (Oda)582-1
Sanskrit-Tibetan Index for the Yogācārabhūmi-śāstra (Yokoyama and Hirosawa)

One distinctive feature that you will notice in this example, that one does not yet see in standard reference works, is that attribution is not simply given for the entire entry as a unit: responsibility is acknowledged for each segment (XML node) of the entry--and as often as possible, with the equivalent Sanskrit or Tibetan. Both characteristics are especially helpful for those who are doing research and translation. Of course, using XML like this, we can display more detailed information if we want to. But whether or not we decide to display it when we publish the HTML files, the users have the option of viewing the XML source data if they like. For the above-shown entry, the XML source data looks like this:

<entry ID="b8a00-8aaa" added_by="cmuller" add_date="1997-09-15" update="2003-10-11"
rad="言" radval="07" radno="149" strokes="00">

<hdwd>言說</hdwd>
<pron_list>
<pron lang="zh" system="py" resp="cmuller">yánshuō</pron>
<pron lang="zh" system="wg" resp="cmuller">yen-shuo</pron>
<pron lang="ko" system="hg" resp="cmuller">언설</pron>
<pron lang="ko" system="mc" resp="cmuller">eonseol</pron>
<pron lang="ko" system="mr" resp="cmuller">ŏnsŏl</pron>
<pron lang="ja" system="kk" resp="cmuller">ゴンゼツ</pron>
<pron lang="ja" system="hb" resp="cmuller">gonzetsu</pron>
</pron_list>
<sense_area>
<trans resp="s.hodge">verbal expression</trans>
<sense resp="s.hodge">(Skt. <term lang="sa">vyavahāra</term>; Tib. <term lang="bo">tha
snyad</term>)</sense>
<sense resp="s.hodge">
<trans resp="s.hodge">expresses, recounts</trans>; (Skt. <term lang="sa">abhi-lap*</term>;
Tib. <term lang="bo">brjod par 'gyur ba</term>)</sense>
<sense resp="s.hodge">
<trans resp="s.hodge">expressing, an expression, an utterance</trans>; (Skt. <term
lang="sa">abhilāpa</term>; Tib. <term lang="bo">brjod pa</term>)</sense>
<sense resp="s.hodge">
<trans resp="s.hodge">discourse</trans>; (Skt. <term lang="sa">kathā</term>; Tib. <term
lang="bo">gtam</term>)</sense>
<sense resp="s.hodge">
<trans resp="s.hodge">a figurative designation</trans>; (Skt. <term lang="sa">upacāra</term>;
Tib. <term lang="bo">nye bar 'dogs pa</term>)</sense>
<sense resp="cmuller">Language, <trans resp="cmuller"><term
lang="en">speech</term></trans> (<term lang="sa">vāc</term>), which is one of the three
kinds of permeation of the store consciousness taught in the <title>Mahāyāna-saṃgraha</title>.
<bibl type="canoncite"><cit><xref idref="b651d-5927-4e58-8ad6">攝大乘論
</xref>&#160;<biblScope source="T" div="num.vol.pg-col-line">T
1593.31.117c3</biblScope></cit>
</bibl></sense>
<sense resp="cmuller">The usage of language to teach the dharma (<term
lang="sa">deśanā</term>). </sense>
<sense resp="cmuller">Language as synergistic with the mental realm of phenomenal
differentiation (<term lang="sa">abhilāpa</term>). </sense>
</sense_area>
<dictref>
<dict><title>Iwanami Bukkyō jiten </title><page>239, 293</page></dict>
<dict><title>Bukkyōgo daijiten (Nakamura)</title><page>429b</page></dict>
<dict><title>Ding Fubao</title><page/></dict>
<dict><title>Buddhist Chinese-Sanskrit Dictionary (Hirakawa)
</title><page>1072</page></dict>
<dict><title>Bukkyō daijiten (Mochizuki)</title><page>(v.9-10)1043b</page></dict>
<dict><title>Bukkyō daijiten (Oda)</title><page>582-1</page></dict>
<dict><title>Sanskrit-Tibetan Index for the Yogācārabhūmi-śāstra (Yokoyama and Hirosawa)
</title><page/></dict>
</dictref>
</entry>

IV. Making Contributions

An important future enhancement for the DDB will the development of an input form for contributors, to allow them to readily add new entries, or modify presently existent ones. For the time being however, lacking a formal apparatus for the input of new materials, we have been adding material received in attached mail files, mostly in MS-Word format. As long as the contributors use a format with a uniform structure, and are able to submit their materials in Unicode, using Unicode-mapped diacritics and East Asian characters, there is not that much else that needs to be done, as we are able to do much of the main markup with macros and various scripts. We do, however, encourage users to submit their materials with XML markup to whatever extent they are able to handle it, ranging from a minimal type of markup, up to a fully marked-up document using our DTD. On our web site, we offer users the following options (from *http://www.acmuller.net/ddb/notes/ Basic_Formatting.html*):

Basic Formatting Suggestions for DDB Entries

Topics:

Updated 2004.09.05

A. Introduction

First and foremost, please understand well: **the usage of XML tagging is not necessary for contributing to the DDB. We will happily accept contributions in popular word processor file formats with no XML markup whatsoever.** If, however, you are interested in going a step or two beyond that, and would like to learn something about how we encode our materials, then please read on.

B. Basic DDB Entry Format

Up to now, the basic organization of a DDB entry has been like this (with some abridgments for the sake of simplicity):

Headword: (Han characters)

Pronunciations:

Chinese (Pinyin):
Chinese (Wade-Giles):
Korean (Hangul):
Korean (Ministry of Education System):
Korean (McCuneReischauer):
Japanese (Katakana):
Japanese (Hepburn):
Translation: (Simple, short-phrase equivalent of the headword, if available)
Explanation: (Detailed explanation of the entry headword)

If you were adding a term, you would type the Chinese next to "Headword." You would then add the pronunciations for the languages you know. Someone else can supply the readings for the languages you can't handle. After the pronunciations, we usually make an attempt to offer one (or up to a few) common renderings of the term. If it were a person, place, temple, etc., we would just supply the commonly used name, such as "Zongmi," "Dongshan," "Jinglingsi," etc. If it were a concept, "middle way," etc. This is followed by a detailed explanation, which can have multiple nodes for multiple contributors, as necessary.

Let's look at example. This is an entry regarding the Korean monk Iryŏn. It is an entry for which I provided minimal information many years ago, and which badly needs to be expanded. But its present brevity makes it useful here:

Headword: 一然

Pronunciations:

Chinese (Pinyin): Yīrán

Chinese (Wade-Giles): I-jan

Korean (Ministry of Education System): Iryeon

Korean (McCuneReischauer): Iryŏn

Japanese (Hepburn): Ichinen

Translation: Iryeon

Explanation: (1206-1289) An important Goryeo monk. A prolific writer, who is most famous for his Samguk Yusa [Chinese title here], a collection of facts and anecdotes which is a basic text for the study of the history of Korean Buddhism.

C. Basic DDB Entry Format: XML

Now, for XML. Rather than starting off with an explanation of XML theory, I think it is simpler if I just re-present the above example using a simplified form of XML.

```
<entry>
<hdwd>一然</hdwd>
<pron_list>
<pron>Yiran</pron>
<pron>I-jan</pron>
<pron>Iryeon</pron>
<pron>Iryŏn</pron>
<pron>Ichinen</pron>
</pron_list>

<trans>Iryeon</trans>
<sense> (1206-1289) An important Goryeo monk. A prolific writer, who is most famous for his <title>Samguk Yusa</title> 三 國遺事, a collection of facts and anecdotes which is a basic text for the study of the history of Korean Buddhism.</sense>
</entry>
```

If you look at this for a minute, you will see that there is not much difference between the first example and the XML-tagged example. The basic difference is that here we are using opening and closing tags to delimit information. You will notice that inside the <sense> tags, the title of Iryeon's text, *Samguk Yusa*, is enclosed with the tags <title></title>, indicating that this is the name of written work. We also use similar tags for <term>technical terms</term>, <foreign>foreign words</foreign> and other elements. When this entry is published as HTML, these words will automatically be italicized. We can also use these tags to build indexes.

If can cooperate by using this simple level of XML structuring, it would be greatly appreciated. But once again, it is not absolutely necessary for the task.

D. Basic DDB Entry Format (Fully Developed XML Markup)

The above example shows the barest XML framework—what are called ELEMENT tags. The tags <entry>, <pron>, <title>, etc. are all known as "elements" in XML parlance. But elements can also be enhanced by a very useful secondary layer of information, which is known as ATTRIBUTE information. Please see the same entry, again presented in a manner much closer to the way it is actually contained in our data set:

```
<entry added_by="cmuller" add_date="1990-09-21" update="">
<hdwd>一然</hdwd>
<pron_list>
<pron lang="zh" system="py" resp="c.wittern">Yīrán</pron>
<pron lang="zh" system="wg" resp="cmuller">I-jan</pron>
<pron lang="ko" system="mc" resp="cmuller">Iryeon</pron>
<pron lang="ko" system="mr" resp="cmuller">Iryŏn</pron>
<pron lang="ja" system="kk" resp="cmuller">イチネン</pron>
<pron lang="ja" system="hb" resp="cmuller">Ichinen</pron>
</pron_list>
<sense_area>
<trans resp="cmuller"><person_entry loc="ko">Iryeon</person_entry> </trans>
<sense resp="cmuller"> (1206-1289) An important Goryeo monk. A prolific writer, who
is most famous for his <title lang="ko">Samguk Yusa</title> 三國遺事, a collection of
facts and anecdotes which is a basic text for the study of the history of Korean
Buddhism.</sense>
</sense_area>
```

```
</entry>
```

I believe that the point of most of the attributes should be obvious, but one of the most important that I would like to draw your attention to is that of "resp", which means "responsibility"—thus, "accreditation." Far distinguished from paper publishing counterparts, the usage of XML in a digital reference work allows us to give credit to the person responsible for every small part of the <entry>. Thus, if someone wanted to add another <sense> element (or "node") to this entry, it could easily be done, giving that person credit in the "resp" attribute.

Also commonly used in the DDB is the "lang" attribute, which tells us the language of the text or foreign word that will be italicized. For texts, we also have a "prov" (provenance) attribute. For temples and geographical entries, we have a "loc" (location) attribute. There are a number of others as well.

Using attributes allows for all kinds of programming possibilities, including various font transformations on presentation, creation of detailed indexes, and so forth.

However, once again, for those for whom this is a headache, it is fine if you want to terminate your exposure to XML here. Ensuing discussions will go into a bit more detail on XML for those who are interested, so you may ignore these if you wish.

V. Near-Term Future Prospects for the DDB

During the past year, we have reached a distinctive new stage in the development of the DDB, wherein suddenly a large number of recognized scholarly experts in Buddhist Studies have begun to contribute data, and major university libraries have decided to subscribe. We presently have some 5,000 entries on the queue awaiting input, with new contacts from interested scholars coming weekly. Thus, we appear to be on the verge of being able to declare the DDB project a major success. We still eventually need to figure out a way to round out the balance of the coverage, so that there is more equal representation in terms of sects and cultural traditions, but this can probably be solved by the attainment of another significant grant or two. But if we can continue to grown at the rate of 5-10,000 terms a year for the next five years or so, it will probably be the right time to begin to turn our full attention to the proper completion of the sister project of the

DDB--the CJKV-E dictionary.