社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006－CH－72 (4)
2006／10／27

# 文字オントロジーに基づく文字処理について

守 岡 知 彦†

CHISE project で開発を進めている文字オントロジーに基づく文字処理手法 (Chaon モデル) と環境 (CHISE) の概要について述べる。近年、計算機とインターネットの普及と発展のために、電子テキストの普及が進むとともに、電子テキストの利用法も単に人間が読むためのものから、機械処理可能なデータベースとしての側面が高まって来ている。そこで、本稿では文字の意味論的な側面に焦点を当て、他のソフトウェア・モジュールに対するインターフェイスの側面から文字の表現や処理の問題を考えるとともに、この問題に対する CHISE project における方針について述べる。

# Character processing based on character ontology

MORIOKA TOMOHIKO†

This paper explains an overview of the current state of character processing technology based on character ontology as a result of the CHISE project. Recently, application field of digitalized texts are increasing day by day because of popularization and development of computers and the Internet. Usage of digitalized texts are also increasing and diversifying; now a day, they are not only expected as human readable objects but also machine readable databases. For the current situations of digitalized characters, this paper focuses the issues of character semantics, discusses the issues of character representation and processing for a side of interfaces for other software modules, and describes principles in the CHISE project.

## 1. はじめに

CHISE project で開発を進めている文字オントロジーに基づく文字処理手法 (Chaon モデル) と環境 (CHISE) の概要について述べる。

現在の一般的な文字処理手法である『符号化文字モデル』では、文字に関する知識は文字符号の定義の中に存在し、利用者が勝手に変更することはできない (利用者が勝手に変更すれば情報交換できなくなる) し、そうした知識は計算機の中に存在しない。送り手と受け手で文字 (概念) が同一の符号を共有してさえいれば、計算機中では各文字は単に整数として表現すれば良く、少ない計算資源で文字を処理することが可能であるといえる。しかしながら、同一の符号の形式 (syntax, encoding, etc.) を共有することもさることながら、同一の文字概念・観念を共有することは必ずしも容易なことではない。特に、漢字のように基本となる文字数が多いだけでなく異体字も多く文字の同一性に関する観念が曖昧な場合や、古典文献や古代文字のように用字法や文字の同一性に関する観念が現代とずれがあったり未解明の部分がある場合、あるいは、用字法におけるテキスト (文脈) 依存性が存在する場合、固定的な汎用文字符号の中に安定的な文字観念を同定し各文字を符号化しようとする符号化文字モデルの前提と矛盾することが少なくない。

一方、近年、計算機とインターネットの普及と発展のために、電子テキストの利用分野の多様化が進んでいる。これにより、日常的な文書処理や通信だけではなく、行政 (地名、人名) や学術資料など、多数の異体字を抱える分野においても電子化が盛んになってきている。また、電子テキストの普及とともに、検索やデータ分析をはじめとする利用法の多角化も進んで来ており、電子テキストは単に人間が読むための (表示や印刷さえできれば良い) ものから、機械処理可能なデータベースとしての側面が高まって来ているといえる。特に漢字の場合、異体字が多いため、単純な検索であっても、異体字処理が必要となり、そのために、異体字関係の対応表が必要となる。また、文字数が多いため、文字を探すための情報も必要である。こうした目的の他、過去のさまざまな (レガシーな) 文字符号に対処するために文字符号間の対応表 (mapping table) も盛んに利用されている。こうしたことを考えれば、文字に関する知識の機会可読な表現 (文字オントロジー) の整備とそれを利用した文字処理技術の研究・開発は大変重要であるといえる。しかしながら、現状では、符号化文字モデルを前提に必要に応じて ad hoc に限定的な文字データベースが作られることが多

† 京都大学人文科学研究所
Institute for Research in Humanities, Kyoto University

く、本来文字が担い得る情報を十分に活かした処理ができていないように思われる。

我々はこうした現状を鑑みて、符号化文字モデルの前提を一度白紙に戻した上で、計算機上における文字処理を分析し、知識表現的な枠組によって文字を表現する Chaon モデル を提案するとともに、このモデルに基づく文字処理環境 CHISE の実現を目指している。

## 2. 符号化文字モデルの問題点

検索や再利用が容易であるという電子テキストの優れた特徴は符号化文字技術に基づいているが、これはもともと英語用ラテン文字のために設計されたものであるといえる。さまざまな字種に対して符号化文字技術を適用する試みは古くから行われてきたが、英語用ラテン文字との差異が大きい用字系 (script) ではしばしば技術的な齟齬に悩まされてきた。そうした齟齬は符号化文字技術の原動力となり、この技術を豊かにしてきたが、それでも齟齬が無くなった訳ではなく、また、本来単純であった符号化文字技術の複雑化も招いている。

漢字においても、まず、文字数の多さが問題になり、その後、大規模文字集合の登場により、何を文字と考え何を符号化するかとか、文字の同一性をどう考えるかなど、符号化文字技術を支える抽象文字概念に関する根源的な問題が顕在化した。そして、文字が増えても依然として外字を必要とする局面は残っている。そして、インターネットの発達によって、各々の外字を大域的に利用可能にするために標準化しようとする欲求が高まったといえる。その結果、符号化文字集合の標準を拡大しようとする動きは際限なく続いている。このような状況により、文字概念に起因する根源的な問題はむしろ拡大しているといえる。

計算機における文字表現に対する不満はこれまでも表明されることはあったが、多くの場合、特に、漢字の場合、その不満は文字符号の拡張や『新たな文字符号体系』の提案という方向に向かった。しかしながら、多数の異体字を収録した文字符号は表示のためには良くても検索には不便である。また、多数の異体字を収録可能な文字符号系の多くは単に符号位置が追加できるという文字符号表現の統語論的問題を解決しているに過ぎず、符号化文字の意味論的問題は無視していることが多い。符号化文字モデルにおいては符号化文字の意味論は符号化文字集合の定義の中にしか存在し得ないので、単に符号位置が拡張可能であるというだけでは文字に関わる多様な問題の解決につながっていないといえる。

## 3. 従来の文字データベースと漢字辞書の比較

説文解字、康熙字典、あるいは、大漢和辞典といった字書・字典は現在の文字符号の重要な典拠の一つともなっており、いまなおこれらの電子化以前の漢字辞書の重要性は失われていない。

この理由の一つは文字の表現の問題である。ISO/IEC 10646-2:2000[2] で制定された統合漢字拡張 B の登場まで、標準的な汎用文字符号の収録文字数は康熙字典や大漢和辞典の収録文字数に届かなかった。よって、符号化文字によってこれらの漢字辞書を十分に表現することがそもそもできなかったといえる。また、符号化文字は漢字の字体や字形を細かく規定するものではないので、安定的に字体や字形を表現するのにも問題があった。こうしたことから、文字符号を担保するものとしての紙媒体の漢字辞書が不可欠であったといえる。

こうした収録文字数や字体・字形にまつわる問題が技術的に解決できるようになった現在においても、依然、紙媒体の漢字辞書の一般的な文字データベースに対する優越性は残っている。それは字書・字典の収録する情報の種類の多さ、多目的性である。

現在、一般的に文字データベースはなんらかの想定する処理のために構成されている。例えば、マッピング・テーブルならばコード変換のためであり、異体字シソーラスならば異体字あいまい検索などのためである。そうした目的を実現するために必要最小限のデータを用意するのが通常である。このこと自体は当然のことであるが、現在の計算機上での一般的な文字処理の内容がコード変換や異体字あいまい検索といった極めて限定された内容でしかないために一般的に文字データベースの内容もまた極めて限定された内容しか必要としない訳である。

これに対し、字書・字典は漢字の形・音・義の情報や別の文字との関係（異体字・類字関係等）を載せるとともに、場合によっては（近代的な辞書の場合、特に）典拠情報や典拠毎の異説、用例等も掲載されている。これは字書・字典の用途の多目的性を反映しているといえる。即ち、文字の形や発音や意味を調べるのにも使われるということである。また、単にある文字を調べるために使われることもあるが、古典中国語（漢文）を読むための辞書という側面もあり、用例に関する情報も載っていたりする。

つまる所、電子化以前にできた漢字辞書と一般的な文字データベースの差はそれを使って行いたい作業の差といえる。

## 4. 従来の文字処理の問題点を超克するために

現在の計算機上での一般的な文字処理があまり高度でない（知的でない、紙ベースの辞書をひきながら人間が行う文字処理に機能的にだいぶ負けてる）のは、従来の文字処理が実現しようとした目標が貧しいからではないかと思われる。このことは、そもそも計算機

上での『文字』(符号化文字) やその列 ((符号化) 文字列) によるテキスト表現 (plain text) という観念を天与のものとし、その枠組の下でソフトウェア環境を構築してきたということに関係しているのではないかと思われる。[6]

そこで著者は、巨大な収録文字数を誇る符号化文字集合やエンコーディングを工夫した優れた文字符号を作ろうとするのではなく、符号化文字技術というもの自体に疑いの目を向け、符号化文字技術に依存しない文字表現を模索するようになった。その後、田中哲氏や g 新部裕氏らとの議論を通じ、この問題は次のように整理された：

(1) 脱符号化：画像のような符号化文字列以外のもので記号を表現する
(2) 文字定義：文字は 1 の記号を基に、性質や関係を定義することによって定義する
(3) (カラーパレット的) 局所的文字符号：2 で定義した文字オブジェクトに対応する短い整数値 (符号値) が必要な場合は、局所的に利用する整数値を適当に作る。この際、文字オブジェクトとの対応関係を管理する

そして、1997 年には 1 と 2 に焦点を当てた Lisp 系記号処理モデル である *My Symbolic System* が構想され、1998 年には 2 と 3 に焦点を当てた文字処理モデルである *UTF-2000* モデル が構想された。☆ 1998 年には g 新部氏によって UTF-2000 based on GNU Emacs 20.2.90 が試作された。そして 1999 年には著者によって、現在の XEmacs CHISE の前身である、XEmacs UTF-2000[9][1] の開発がスタートし、*UTF-2000* モデルの具体化が本格化した。この *UTF-2000* モデル が後に Chaon モデルとなるものの原型であり、この開発プロジェクトを発展的解消し、新たな開発者を加えて CHISE project が始まった。

このように、CHISE project は (計算機の中における) 文字の意味論的な側面に焦点を当てたデータベース的な手法に基づく文字処理システムを作ろうというものである。ここで、意味論的な側面から符号化文字モデルの問題点を再度振り返ってみる。

符号化文字モデルにおいて、符号化文字 (符号位置) の意味は符号化文字集合で規定された文字の概念 (抽象文字) であるが、それは実質的には文字処理時における文字の振る舞い方のことだといえる。例えば、表示 (組版) や自然言語処理のように、文字処理を含みつつもそれとは違う領域・モジュールの問題も扱う場合において、文字は対象とする処理において、その問題に関係ない要素を捨象され、対象領域・モジュールにとって必要とされるように十分に区別され (不要な区別が無視され)、適切に振る舞わなければならない。即ち、符号化文字の意味論は文字を利用する (文字処理以外の) 領域・モジュールとの関係によって規定されるべき問題であるといえる。こうしたことを鑑みれば、符号化文字集合の定義は、組版やテキスト処理、あるいは、自然言語処理や意味処理なども含んだ文字言語処理に全体のアーキテクチャを考える中で、文字処理モジュールが引き受けるべき範囲を考え、その上でさまざまな処理を十分に可能とする包摂規準を定義し、各符号位置がカバーする文字の範囲を明確にすることが求められる。

しかしながら、現実には必ずしもそのようにはなっておらず、レイヤ (layer; 層) 化の問題というべき事象が見受けられている。例えば、異体字の問題を符号化文字層で解決するのか、それとも、上位層であるマークアップ層で解決するのかということについては必ずしも一貫した規準は存在しない。また、符号化文字集合の側が一方的に『これはマークアップ層で解決すべき問題である』と看做したとしても、マークアップ層の側が実際にその問題を自分の問題として引き受けなければ現実には問題は解決されない。また、逆に、異体字タグのように、符号化文字層の側がマークアップ層での処理に類似した機構を入れてしまい、2 つのレイヤ間でコンフリクトが生じてしまう場合もある。ただ、現実的には、こうした縄張争いよりも逆縄張争い、即ち、厄介で面倒な問題のレイヤ間での押し付け合い状態になる方が多くなりがちではないかと思われる。意地悪な言い方をすれば、『符号化文字』という概念は、「文字」という概念を適切に定義することにより、文字層で扱いたくない問題をグリフ層だとかマークアップ層だとか自然言語処理層に押しつけるために導入しているように見えなくもない。符号化文字層において完備で整合性がとれた符号を定義できたとしても、文字言語処理全体の中でレイヤの狭間でこぼれ落ちてしまう問題が生じてしまうのだとしたらやはり問題といわざるを得ない。

このように、(符号化文字モデルにおける) 文字に関わる問題の多くは、符号化文字として抽象化される際に捨象されたりして、符号化文字の世界・文字処理モジュールと他の領域・処理モジュールの連係がうまくとれないことに起因していると考えられる。とするならば、こうした問題を解決するには文字処理以外の領域・モジュールと十分にインターフェースがとれるように文字表現・文字処理モジュールをデザインすれば良いといえる。CHISE のテクニカルなデザインはこのような考えに基づいている。

一方、利用者の側から見た時の CHISE project の目標は、JIS X 0208 や Unicode といった汎用文字符号に依存しない文字技術を開発し汎用文字符号に制約されない文字処理環境を実現すること、簡単にいえば『(利用者が思った通りに) 自由に文字が使えるようにすること』である。この『自由に文字が使える』ということは、(さまざまな) 利用者が考える (さまざま

☆ どちらも、g 新部裕氏の命名である。

な）文字観念が表現・処理できることだと考えられる。このことは、言い替えれば、文字に関する知識処理環境を実現することだといえる。

文字は第一義的には社会で共有される（言語的）情報交換のための道具であるから、その中心には社会的に共有される文字観念を記述した文字オントロジーを据えるのが妥当であると思われる。さまざまな文字観念をバラバラに記述することは理念的には可能なことではあるが、それらをどんどん集めていくことは記述量の爆発を招く。この問題はオブジェクト指向における継承のような差分的な記述を行うことである程度解決可能であると思われる。これはもっとも近い文字（観念）との差異を記述することである。この差異は文字（観念）間の関係と捉えることができ、その集合を可視化することで文字（観念）間の関係の見取図を与えることができる（図 3）。

『社会的に共有される文字観念』は曖昧で恣意的な概念であり、そうしたものに立脚して固定的な文字の符号化を行うことが符号化文字モデルの問題のひとつである訳であるが、文字観念を記述した文字オントリジの利用において、継承された（より共有された）文字観念と継承した（よりローカルな）文字観念がシステム的に同等に扱えるのであれば、言い替えれば、文字知識データベースの差分的記述等の実装（構成法）が利用者にとっての視点から隠蔽されていれば、曖昧で恣意的な概念に制約されるという問題は回避できるといえる。

また、何を『(より共有された）文字観念』と看做すのかという点に関しても、文字オントロジーの記述という視点によってその規準を与えることができる。即ち、全体の記述量を最小にできるような文字観念間の継承関係のグラフを得た時、ハブとなるのが『共有された文字観念』といえる。もちろん、この世に存在し得る全ての文字観念の集合全体を書き尽くすことができないことや全体の記述量を最小にできるような文字観念間の継承関係のグラフが一意に決まるかどうか判らないことを考えればこの『規準』は多分に理念的・理想的なものであるが、とはいえ、実際に必要とされる有限の文字観念の集合を表現した文字オントロジーを構成する上でもこの視点は適用できるといえる。

## 5. CHISE における文字表現

### 5.1 Chaon モデル

CHISE では『Chaon モデル』と呼ぶ方法によって文字を表現するようになっている。これは汎用符号化文字集合に依存することなく自由に文字を表現するために我々が提案しているもので、表現したい文字に関する知識（文字の性質の集合）の機械可読な表現によって文字を表現し操作する方法である。

Chaon モデルでは、文字を説明するための要素（文字の性質や用例など）を『文字素性』(character feature) と呼ぶ。文字素性としては、部首、画数、部品の組合せ方に関する情報（漢字構造情報）、発音、意味、用例、その他文字処理で必要となる各種情報などが考えられる。

Chaon モデルにおける文字表現は文字素性の集合であり、文字に関するさまざまな要素を文字素性として記述可能である。また、対象とする文字を 1 つの文字素性だけで記述することも可能であるし、多数の文字素性を用いて非常に詳細に記述することも可能である。なお、Chaon モデルでは文字と文字の集合は本質的に区別されない。符号化文字モデルでは、通常、文字符号の仕様（規格）が想定する抽象文字を文字とし、抽象文字はある範囲の字体を包摂するものとなっている。ここで、文字か字体かが必然的に判別可能なものであるなら問題はないのであるが、実際にはこれは文化的なものに依存しており、社会的に共有される規範や文字観念はあるにせよ、必ずしもかっちりと線を引けるものではないといえる。そして、これらは歴史的・地域的に変化することがあり、用途や文脈にも依存し得る。よって、Chaon モデルでは、技術的な枠組としては、抽象文字、字体、字形のような文字の抽象度に関する固定的な線引きを行わない。ある文字素性の集合を字形を表したものと思うか、字形の集合である字体を表したものと思うか、字体の集合である文字を表したものと思うか、はたまた文字の集合であると思うかは、その文字素性の集合の解釈に依存し、Chaon モデルはその解釈を規定しない。それは利用者の自由である。

### 5.2 文字間の関係の記述

Chaon モデルは文字素性の集合として文字（の集合）を表現（指示）する手法であり、そのモデルそのものには文字間の関係で構成されるネットワーク構造を表現する仕組みは存在しない。しかしながら、文字間の関係を表す文字素性を導入し、その値として文字（の集合）（の集合）を取るようにすれば、文字間の関係で構成されるネットワーク構造を記述することができる（図 3）。ここで文字間の関係を表す文字素性のことを関係素性と呼ぶ。

CHISE では関係素性を表すのに *->foo* と *<-bar* という先頭 2 文字が -> または <- とする文字素性名を用いることにしている（但し、5.3 節で述べる文字素性メタデータ名は除く）。ちなみに、これは矢印を表現したものである。こうした文字素性名のことを関係素性名と呼ぶことにする。

関係素性の性質は次のように定義されている：

文字 A, B が存在する時、A の文字素性 (*->foo* ... B ...) は、関係 A *->foo* B が存在することを意味する。この時、文字 B には逆関係を表す文字素性 (*<-foo* ... A ...) が存在する。

関係素性の値には文字の集合（リスト）を取ること

になっており、(->*foo* B C D) などのように書くことができる。この場合、定義される文字を A とする時、関係 A ->*foo* B と関係 A ->*foo* C と関係 A ->*foo* D が存在することになる。

### 5.3 階層的素性名方式

汎用的な文字データベースを作る場合、用途や立場・学説などによって、文字素性の値に複数の選択肢を設けたくなる場合がある。こういう時、単純に複数の値が記述できるだけでなく、各々の値の出典情報などのメタデータも付加したいことが少なくない。こうした場合、文字素性の値か名前のどちらかを構造化する必要がある。『階層的素性名方式』というのは後者の方法の一種である。

階層的素性名方式は構造化の対象となる文字素性の名前（文字素性基底名）に値を選択するための識別子（『ドメイン識別子』と呼ぶ）を付けた文字列を生成し、それを名前（文字素性具象名）として用いたり、同様にメタデータ識別子を付けた文字列を生成しそれを名前（文字素性メタデータ名）として用いる方法である。この名前は次のような規則で生成される：

文字素性具象名
:= 文字素性基底名 @ ドメイン識別子

文字素性メタデータ名
:= 文字素性具象名 * メタデータ識別子
| 文字素性メタデータ名 * メタデータ識別子
| 文字素性メタデータ名 @ ドメイン識別子

ドメイン識別子
:= 基底ドメイン識別子
| ドメイン識別子 / 基底ドメイン識別子

例えば、総画数を表す文字素性名を `total-strokes` とし、ドメイン識別子として `ucs` を用いる時、文字素性具象名は `total-strokes@ucs` となる。また、出典情報を表すメタデータ識別子を `sources` とする時、`total-strokes@ucs` の出典情報は

`total-strokes@ucs*sources`

で表される。

部首と部首内画数のように異なる種類の文字素性の値が対応関係を持っている場合、ドメイン識別子を用いてその対応関係を表すことができる。例えば、部首を `ideographic-radical`, 部首内画数を `ideographic-strokes` で表す時、

`ideographic-radical@ucs`
`ideographic-strokes@ucs`

の両者は対応する。

値を構造化する手法と名前を構造化する手法を比べた場合、前者はドメイン識別子を必要としないという利点を持っているものの、値が構造データとなるので C のような単純な記憶管理機構しかない環境では不便である。また、高速化を要するような単純な処理の場合、大抵、複数の値やメタデータを必要としないといえる。また、Chaon モデル的には文字が文字素性の単純な集合になっている方が自然であり便利であるが、値を構造化すると複数の値同士の集合演算を要し、処理が複雑になる。このようなことを鑑み、現在の CHISE では共有文字データベース内では原則として値ではなく名前を構造化する方針を採っている。

## 6. 文字オントロジーの構成法

Chaon モデルはメタなモデルであり、これに基づく具体的な文字データベースの構成法や文字処理システムの実現の仕方にはさまざまな形がありえる。このことは文字に関するさまざまな観念や概念を記述可能であることを意味しているといえるが、とはいえ、現実に文字を処理するためには、具体的なアーキテクチャや文字知識を記述するための指針が必要だといえる。

そこで、現在の所、CHISE project では『(社会的に) 共有される文字観念』に相当する文字知識を記述した文字オントロジー（共有文字データベース）を中心にした文字処理アーキテクチャを採用している。この文字オントロジーとして「CHISE 汎用文字データベース (CHISE-DB)」の編纂を続けている。

「CHISE 汎用文字データベース」はさまざまな文字観念を差分的に記述するためのベースになるような汎用的な文字データベースを提供することを目指している。このため、字形・字体の細かな差異を捨象した抽象的文字観念、Unicode や JIS X0208 のような各種符号化文字集合における符号化文字、さまざまな文字符号の規格や辞書などの文字表における例示字体・字形の情報など、抽象・具象のさまざまなレベルの代表的文字概念を収録している。

これらの各レベルは主に字体差（比較的大きな形の差異）を示す->*denotational*, <-*denotational* 素性と比較的小さな字体差・字形差（比較的小さな形の差異）を示す->*subsumptive*, <-*subsumptive* 素性を用いて、文字オブジェクト間の継承関係として記述している。即ち、

抽象的->*denotational* 具象的

や

抽象的 ->*subsumptive* 具象的

という風に文字間の継承関係を記述する訳である。->*denotational* と ->*subsumptive* は混在して使うことができ、

大粒度抽象文字 ->*denotational* 中粒度抽象文字->*denotational* 細粒度抽象文字 ->*subsumptive* 字体->*subsumptive* 抽象字形

などのように多段的継承関係を記述することも可能である。

漢字の場合、抽象・具象関係は形・音・義のそれぞれに存在し得るといえるが、今の所、基本的に形の側面に基づくものしか扱われていない。しかしながら、他の要素に基づく抽象・具象関係も将来的には記述したいと考えている。形・音・義それぞれの抽象・具象関係が一致しない場合でも、5.3 節で述べた階層的素性名方式に基づき、それぞれを別ドメインとすることで多元的な継承関係を記述することができる。この場合、どのドメインの継承関係を用いるか（あるいは用いないか）、優先させるかといったことはアプリケーションとドメインを対応づけることによって制御可能である。

## 7. CHISE システムの現状

Chaon モデルに基づく文字処理は、文字を符号ではなく文字素性の集合として扱い、文字素性によって操作するものとなる。このことは、Chaon モデルに基づく文字処理系が文字オントロジーを操作するためのある種のデータベース・システムとなることを意味する。

このため、CHISE Project では、CHISE 環境で文字情報を共有するための文字データベース処理系およびそのコンテンツ、そして、文字データベースを参照して文字処理をアプリケーションの開発を行っている。

具体的に、

**libchise** 文字データベース操作のための基本機能を提供するためのライブラリ

**XEmacs CHISE** XEmacs[5] に基づく Chaon 実装（Emacs Lisp 処理系および対話型編集環境）

**Ruby/CHISE** Ruby[4] に基づく Chaon 実装（スクリプト言語）

**Perl/CHISE** Perl に基づく Chaon 実装（スクリプト言語）

**Ω/CHISE** 多言語 *TeX* 処理系 *Omega*[3] に基づく Chaon 実装（組版システム）

**Kage** 漢字グリフ合成システム[8]

といったシステムの開発が行われている。

XEmacs CHISE, Ruby/CHISE, Perl/CHISE は CHISE 環境の文字データベースを読み書きすることができる。また、Ω/CHISE は参照のみを行い処理を行う。

libchise は現在では **libconcord** という Concord システム[10] の基本機能を提供するライブラリによって書き直されている。この Concord は CHISE が文字だけを対象としているのに対し、任意のオブジェクトを対象とできるよう一般化したものである。これにより、文字以外の対象も Chaon モデル風に表現・操作できるようになり、将来的には文字を含むさまざまなオブジェクトを素性の集合として表現したり、オブジェクト間の関係のネットワークとして捉えたりできるようにし、文字レベルの処理から自然言語処理、意味処理などをシームレスに統合するための基盤を与えようとしている。

## 8. WWW アプリケーション

Chaon モデルに基づく統合型文字処理環境としては XEmacs CHISE が存在するが、これは現在の所、Linux や Mac OS X といった Unix 系 OS 上で動作し、これらの環境に XEmacs CHISE および関連パッケージ、フォント等をインストールすることで利用可能になるが、現在普及している Windows 系の OS では CHISE の情報は利用可能でなかった。このため、2005 年から、WWW 上で CHISE の機能を提供する試みも行っている。

### 8.1 CHISE IDS 漢字検索

これは CHISE 文字データベース中の漢字構造情報の検索機能を WWW 上で提供するサービスである。「部品文字列」窓に 1 つまたは複数の部品を指定し、漢字を検索することができる。

検索を実行すると、指定した部品列の各部品を少なくとも 1 個は含んでいる漢字の一覧が表示される。ある文字を部品とする漢字が存在する場合、その文字の下にインデントして木構造状に表示される。

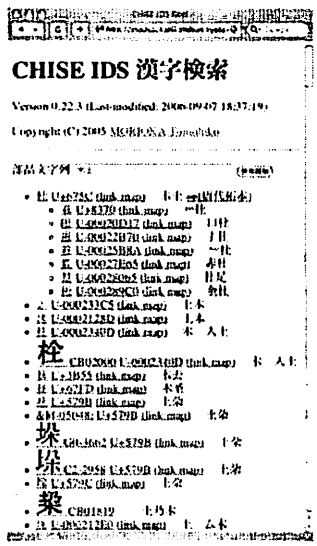


図 1 CHISE IDS 漢字検索

### 8.2 文字の情報

「CHISE IDS 漢字検索」の検索結果を示す各行の一番左にある、文字の項目をクリックすると、その文字に関する情報を表示する画面を見ることができる（図 2）。

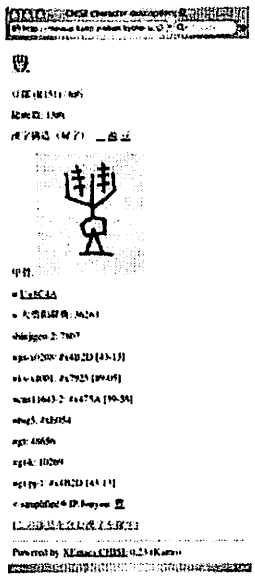

図 2　文字に関する情報の表示画面

この中には漢字構造情報や異体字情報やその他関連する文字へのリンクが存在する場合があり、こうしたリンクを辿ることで関連字の情報を見ることができる。

特に、漢字構造情報は入力しづらい漢字部品を探す上で有用である。例えば、「辶」(辵) を含む漢字を探したい時に、まず「進」で検索して、「進」の文字情報画面を開き、その「漢字構造 (解字)」の項目にある「辶」をクリックすれば、「辶」の文字情報画面を開くことができる。その後、画面下方にある「この部品を含む漢字を探す」というリンクを押すと、「辶」を含む漢字の一覧を得ることができる。

### 8.3 CHISE 漢字連環図

「CHISE IDS 漢字検索」検索結果を示す各行の左から 3 番目にある、(link map) という項目は、上地宏一氏による「CHISE 漢字連環図」(chise_linkmap)[7] というサービスに対するリンクになっている。

これは 5.2 節で述べた文字間の関係を可視化するサービスであり、異体字・類字関係を概観することができる。

## 9. おわりに

現在の一般的な文字処理手法である『符号化文字モデル』における文字処理の問題点について議論するとともに、主に文字処理という視点から文字表現の問題について考え、CHISE project における方針について述べた。

現在の豊富な計算資源を前提に今後の高度なテキスト処理の実現を目指すためには、文字処理の世界においてもこうした処理を支援するためのインターフェイスを用意することが重要であると考えられ、そうした観点に立った文字表現や文字処理のアーキテクチャを考えて行くことが重要である。

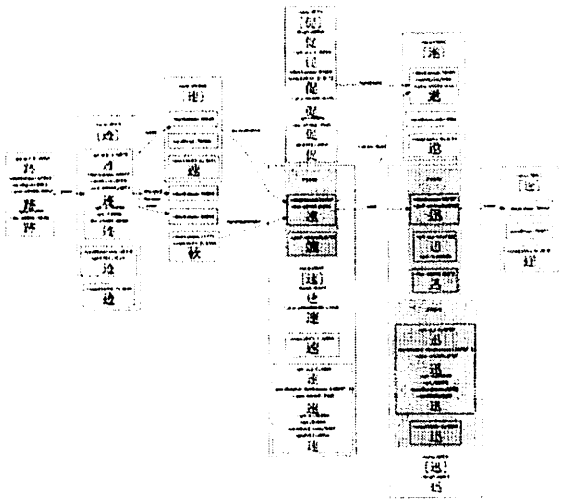

図 3　「CHISE 漢字連環図」の表示画面 (U+8FC5)

CHISE project における文字表現手法である **Chaon** モデルは文字を素性の集合によって表す、言い替えれば、マークアップ的な発想で文字を表現する手法である。Chaon モデルでは利用者やアプリケーションが必要とするインターフェイスに対応する文字素性を定義することで文字処理モジュールと他の領域のインターフェイスが拡張しやすくなるようにデザインされている。文字素性の集合が文字の定義になるので、アプリケーションに依存した文字の区別を導入可能であり、新たに定義された文字は与えられた文字素性や既存の文字との関係の情報によって十分にその意味論的側面を規定することが可能であり、単なる見えるだけの外字や大規模グリフセットではなしえないことを実現可能であるといえる。

CHISE の可能性を発揮するためには文字オントロジーの拡充、従来型システムのためのインターフェイスの拡充、CHISE 的な発想に立った新たなテキスト処理技術の拡充等、さまざまな課題が存在するが、現時点でも一定の有用性が確認できる。今後はこうした課題を解決して行くとともに、CHISE の仕組を一般化した Concord[10] を用いて、文字と関係付けられた文字以外の情報に関する知識の表現や処理に関しても取り組んで行きたいと考えている。

## 参考文献


1) bit 別冊「インターネット時代の文字コード」, 第 9 章「文書編集系における文字コード」. 共立出版, 2001.

2) 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.
3) The omega typesetting and document processing system. http://omega.cse.unsw.edu.au:8080/.
4) The object-oriented scripting language Ruby. http://www.ruby-lang.org/.
5) XEmacs. http://www.xemacs.org/.
6) 安岡孝一. 紙テープの呪縛. シンポジウム「文字情報処理のフロンティア 過去・現在・未来」予稿集. 花園大学 国際禅学研究所, Jun 2004.
7) 上地宏一. CHISE 漢字連環図. `http://fonts.jp/chise_linkmap/`.
8) 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, Vol.3, pp. 143–147, 2002.
9) 守岡知彦. UTF-2000 — 汎用文字符号に依存しない文字表現系の展望. アジア情報学のフロンティア — 全国文献・情報センター人文社会学学術セミナーシリーズ No.10, 全国文献・情報センター人文社会学学術セミナーシリーズ, 第 10 巻, pp. 13–24, 2000.
10) 守岡知彦. Concord: プロトタイプ方式のオブジェクト指向データベースの試み. Linux Conference 抄録集, Vol.4, , 2006.