

情報通信基盤としての文字処理環境の整備
に関する研究

池田 佳代

電気通信大学大学院電気通信学研究科

博士（学術）の学位申請論文

2010年3月

情報通信基盤としての文字処理環境の整備
に関する研究

博士論文審査委員会

主査	兼子	正勝	教授
委員	福田	豊	教授
委員	吉浦	裕	教授
委員	太田	敏澄	教授
委員	高橋	裕樹	准教授
委員	坂本	真樹	准教授

著作権所有者

池田 佳代

2010

A Study on Improvements of Character Processing Environment as ICT Infrastructure

IKEDA Kayo

Abstract

In information and communication technology (ICT), a character processing environment for the input and output of characters is indispensable as information infrastructure. Japanese language processing environment to input and output Japanese characters is essential for most ICT research projects. Many research projects are now underway that take this infrastructure as a given.

However, using Japanese requires the capability to input more than 10,000 characters, and a character processing environment unique to Japanese is necessary to output the characters. Therefore, many challenges remain even today; one problem is the absence of open Japanese fonts.

With calls for internationalization accompanying the spread of the Internet, the Japanese government has been promoting open source software (OSS) to

increase the competitiveness of Japan's ICT industry. However, Japanese fonts are necessary for Japanese language processing and displaying Japanese characters in development environments for OSS; Japanese fonts are also necessary for multi-OS environments. However, no Japanese fonts are currently available for use with OSS. The absence of open Japanese fonts has been a formidable barrier to OSS usage and the Japanese software industry. Moreover, considering the Japanese character processing environment as infrastructure for Japanese society, the "unsupported characters problem" adversely affects how kanji characters for names of people and places are handled on networks. When digitizing the Japanese Basic Resident Register network system, textbooks, and other educational material, a solution to the "unsupported characters problem" is essential for dispelling the notion that digitization could bring about a loss of identity or culture.

Through advances in IT, we have reached an age where multilingual and multi-character set information resources are being accumulated worldwide and

transmitted over the Internet. At the dawn of the computer age, the characters that could be handled by computers were centered around Latin characters. Moreover, only a small number of characters could be handled, as those represented by 7-bit ASCII character code. However, the international standards, ISO/IEC10646 Universal Multiple-Octet Coded Character Set and Unicode, were created in attempts to uniformly handle the languages used across the world by using the development of IT environments that support multiple languages, and the spread of the Internet. This has established a base on which multilingual and multi-character set information resources can be accumulated using a character encoding shared throughout the world. However, since the reality is that the methods available to actually access multilingual and multi-character set information resources are limited to particular languages and characters, language, character, knowledge, and culture barriers exist.

Our goal was to prepare a Japanese and multilingual character processing

environment for Japan's information infrastructure.

In conducting this research, we first surveyed existing Japanese character processing environments and clarified existing problems.

Moreover, regarding Japanese public fonts necessary for the information infrastructure, we enumerated the conditions by which changes in technology and the social environment have made a new domain necessary. We also considered examples of open fonts for Japanese, and discussed future challenges for Japanese public fonts. We also researched licensing for Japanese public fonts that support the information infrastructure.

Characteristics of fonts include elements of programming and content. However, fonts also have unique elements, and Japanese fonts have elements specific to Japanese that must be preserved. Here, we consider a new license from the perspective of Digital Rights Expression (DRE), which is most appropriate for open Japanese public fonts, and is a digital asset that supports the information infrastructure.

To address the “unsupported characters problem problem”, we considered information exchangeability and interoperability. We defined a glyph database to handle the kanji characters as character figures and proposed a method of searching for kanji characters in the glyph database. We also examined how glyphs in search results should be used in the societal infrastructure.

Next, we examined existing multilingual character processing environments and clarified existing problems. We also aimed to allow multilingual information sources to be accessed by using a text input method unrestricted by OS or target language, and we researched an input support system with multilingual support. We also aimed to enable users who are still learning a language and not familiar with vocabulary to be able to easily access information sources to find the word that they are seeking. For this purpose, we had inexperienced users use the proposed system and we conducted evaluation experiments. Finally, we implemented an input support system that supports multiple languages, and demonstrated an input environment that supports multiple Japanese characters

by performing experiments with ideographic variation indication using Ideographic Variation Sequence (IVS), which is standardized for Unicode.

This research has contributed to defining a Japanese public font with the high interoperability necessary for a Japanese character processing environment in ICT infrastructure. We have also developed a Japanese public font license from the perspective of DRE and received Open Society Initiative approval. Thus, we were able to prepare an environment in which Japanese fonts can be freely circulated as digital assets.

The greatest challenge for multilingual character processing environment as information infrastructure for Japan is the method for accessing multilingual information. Therefore, we have provided a multilingual input method for accessing multilingual information. This presents an important opportunity to prepare character processing environments for Japanese and multilingual characters as information infrastructure.

情報通信基盤としての文字処理環境の整備 に関する研究

池田 佳代

概要

ICTの中で文字を入力し出力する文字処理環境は、情報通信基盤として欠かせないものといえる。我が国の情報通信基盤として、多くのICT研究において日本語の文字を入力し出力するための日本語処理環境は欠かせない存在であり、その基盤を前提に推し進められている研究が多数存在するといえる。しかし、コンピュータ上で日本語を扱うということは、1万字以上の文字を入力し、それを文字という可視的な形として出力するための日本語独自の文字処理環境が必要であるため、現在も、多くの課題を抱えている。

その一つとして、オープンな日本語フォントの不在という問題がある。インターネットの普及により国際化が叫ばれる中、日本のICT産業が競争力をもって開発を推進するため日本政府はオープンソースソフトウェア（以下OSS）普及推進を進めてきた。しかし、OSSにおける開発環境、異なるOS間におけるマルチOS環境で、日本語の処理、表示を行うためには日本語フォントは必須であるが、これまで、OSSで利用可能な日本語フォントは存在していなかった。オープンな日本語フォントの不在がOSS活動さらには日本のソフトウェア産業の活性化の大きな障壁となっている。

また、我が国の社会基盤として日本語の文字処理環境を見た場合、人名や地名などの漢字をどのようにネットワーク上で扱うかといういわゆる外字問題が存在する。住民基本台帳ネットワークシステムや、教科書等の教育用コンテンツのデジタル化にあたり、デジタル化がアイデンティティや文化の喪失を招くという印象さえ与えかねない現状に対し、外字問題への対応が期待されている。

一方、ICT化の進展と共に、世界中で多言語・多文字情報資源が蓄積され、インターネット上で発信される時代となった。コンピュータ上で扱うことのできる文字

は、コンピュータ黎明期にはラテン文字が中心であり、ASCIIにおける7ビット文字コードに代表されるように、少ない文字数しか扱うことができなかった。しかし、各国語対応のICT環境の進展と、インターネットの普及により、世界中の言語に用いるあらゆる文字を統一的に扱おうとする国際標準規格:ISO/IEC10646 Universal Multiple-Octet Coded Character Set および Unicode が規格化された。これによって、世界中で共通した文字コードにより、多言語・多文字情報資源を蓄積する基盤が整備されてきた。

しかしながら、これらの情報資源に対して実際にアクセスするなどの活用手段は、特定の言語および文字に限定されているのが現実である。ここには、言語、文字、知識、文化の「壁」が存在する。ユーザーにとって限られた既知の言語や既知の文字が使用されている情報資源のみにアクセスが集中している。

本研究の目的は、我が国の情報通信基盤として、日本語および多言語の文字処理環境を整備するために、いくつかの必要な貢献をすることである。

本研究では、まず日本語文字処理環境の研究として、日本語文字処理環境を概括し、その問題の所在を明らかにした。その上で、情報通信基盤として必要とされてきた日本語パブリックフォントについて、技術と社会環境の変化により新たなドメインが必要とされてきた状況を整理し、IPAフォントでのオープンフォントの整備事例を提示し考察し、今後の日本語パブリックフォントの課題を提示した。

さらに、情報通信基盤を支える日本語パブリックフォントのライセンスの研究を行った。フォントは、その性格上プログラムとコンテンツの両方の要素を含んでいるが、加えてフォント特有の要素もあり、かつ日本語フォントは日本語特有の保護すべき要件を備えている。本研究では、情報通信基盤を支えるデジタル財の1つであるオープンな「日本語パブリックフォント」に最適なライセンス形態について、デジタル著作権表明(Digital Rights Expression: DRE)の視点に立脚し検討を行った。

そして、情報通信基盤としての外字問題への対応を、情報交換性や相互運用性といった観点から検討し、漢字を文字図形として扱うグリフデータベースを定義し、グリフデータベースからの漢字の検索方法や検索したグリフを社会基盤の中でどのように利用するかについての提案を行った。

次に、多言語・多文字処理環境の研究として、多言語・多文字環境の概括と問題

の所在を明らかにした。その上で、多言語情報資源へのアクセスを、OS や対象言語の制約を持たない文字入力手段の提供により実現することを目指し、多言語に対応した入力支援システムの研究を行った。特に、利用したい言語に不慣れな言語学習途上のユーザーでも簡単に所望の言語の情報資源にアクセスできることを目指し、実装したシステムについては被験者による評価実験を行った。

最後に、多言語に対応した入力支援システムを利用して、日本語の多文字に対応した入力環境の実証として Unicode で規格化された Ideographic Variation Sequence (IVS) 対応による異体字表示の実験を行った。

本研究の貢献は、ICT 基盤としての日本語文字処理環境に必要とされる相互運用性に富んだ日本語パブリックフォントの定義付けを行い、さらに、日本語パブリックフォントライセンスを DRE の視点に立脚して構築し OSI 承認を得たことにある。結果として日本語フォントをデジタル財として自由に流通できる環境を整えることができた。

また、我が国の情報通信基盤としての多言語文字処理環境の第一の課題を、多言語情報にアクセスするための手段にあると捉え、Web 上の多言語情報にアクセスするための多言語入力手段の提供を行ったところにある。

これにより、情報通信基盤としての日本語および多言語・多文字における文字処理環境整備へ重要なきっかけを与えるものである。

目次

第一章 序論	1
1. 1 本研究の社会的背景	1
1. 1. 1 日本語文字処理環境	1
1. 1. 2 多言語・多文字処理環境	3
1. 2 本研究の学術的背景	4
1. 2. 1 文字処理とは	4
1. 2. 2 日本語文字処理環境	10
1. 2. 3 多言語・多文字処理環境	11
1. 3 本研究の目的と方法	12
第一部 基盤としての日本語文字処理環境	15
第二章 日本語文字処理環境の概括と問題の所在	16
2. 1 はじめに	16
2. 2 世界における日本語	16
2. 3 印刷技術の中の日本語	18
2. 4 コンピュータの中の日本語	21
2. 5 日本語文字入力環境	22
2. 6 日本語フォント	23
2. 6. 1 フォントフォーマットの変遷	23
2. 6. 2 フォントのモジュール化	25
2. 6. 3 日本語フォントの品質	26
2. 7 文字コード	28
2. 7. 1 JIS X 0208	28
2. 7. 2 JIS X 0213	29
2. 7. 3 ISO/IEC10646 と Unicode	32
2. 8 日本語環境の問題	33
第三章 日本語パブリックフォントに関する検討	36

3. 1	はじめに	36
3. 2	フォントのモジュール化・オープン化	37
3. 3	日本の OSS 政策とフォント	38
3. 3. 1	OSS の社会的役割	38
3. 3. 2	日本の政策の中の OSS 推進	40
3. 3. 3	OSS の産業育成に果たす役割	42
3. 3. 4	OSS 推進におけるフォントの役割	42
3. 4	海外における国家レベルでの OSS とフォント政策	44
3. 4. 1	中華人民共和国（中国）	44
3. 4. 2	台湾	45
3. 4. 3	大韓民国（韓国）	46
3. 4. 4	その他	46
3. 5	IPA フォント公開の経緯	46
3. 5. 1	IPA フォント（Ver.001）公開の経緯	46
3. 5. 2	IPA フォント（Ver.002）公開の経緯	48
3. 5. 3	IPA フォント（Ver.002）の使用許諾の定義	49
3. 6	IPA フォント公開の効果	50
3. 6. 1	一般ユーザーからの反響	50
3. 6. 2	Linux デイストリビュータからの反響	52
3. 7	考察	52
3. 8	おわりに	57
第四章	日本語パブリックフォントライセンスの研究	59
4. 1	はじめに	59
4. 2	DRE とは	59
4. 2. 1	デジタル財と著作権	59
4. 2. 2	デジタル権利管理技術（DRM）	60
4. 2. 3	デジタル著作権表明（DRE）	60
4. 2. 4	日本語パブリックフォントと DRE	62
4. 3	フォントの法的保護の現状と問題の所在	63
4. 3. 1	フォントの法的保護の現状	63

4. 3. 2	問題の所在	67
4. 4	フォント向けライセンス	68
4. 4. 1	フォント向けライセンスの現状	68
4. 4. 2	フォント向けライセンスの考察	73
4. 5	日本語パブリックフォントライセンスの確立	74
4. 5. 1	日本語パブリックフォントライセンスのための要件	74
4. 5. 2	IPA フォントライセンスの検討	76
4. 5. 3	OSI 承認プロセス	80
4. 6	おわりに	83
第五章	グリフデータベースの構築	85
5. 1	はじめに	85
5. 2	異体字・外字への取り組みの必要性	85
5. 3	異体字・外字ソリューション	87
5. 3. 1	異体字	87
5. 3. 2	外字	90
5. 4	グリフデータベースの提案	91
5. 4. 1	メタデータの検討	96
5. 4. 2	グリフフォーマットの検討	101
5. 5	グリフデータベースの実装	105
5. 6	グリフデータベースの試用と評価	111
5. 6. 1	グリフデータベースの試用	111
5. 6. 2	グリフデータベースの評価	116
5. 7	おわりに	117
第二部	多言語・多文字処理環境の整備のために	119
第六章	多言語文字処理環境の現状	120
6. 1	はじめに	120
6. 2	ICT における多言語環境の現状	121
6. 3	多言語文字処理研究の状況	124
6. 3. 1	多言語間の連携	124

6. 3. 2	多言語文字入出力	126
6. 4	多言語文字処理の現状と課題	128
第七章	多言語 InputMethod の実装と評価実験	133
7. 1	はじめに	133
7. 2	問題の所在と目的	133
7. 3	提案システム	134
7. 3. 1	多言語対応	134
7. 3. 2	語彙辞書	136
7. 3. 3	インクリメンタルサーチ	136
7. 3. 4	詳細情報表示（支援機能）	137
7. 4	システム実装	137
7. 4. 1	システム構成	137
7. 4. 2	システムの特徴	138
7. 4. 3	変換辞書データ	139
7. 4. 4	ユーザーインターフェース	140
7. 4. 5	クライアント・サーバー間の通信	143
7. 5	評価	144
7. 5. 1	ユーザーによる入力実験	145
7. 5. 2	結果	150
7. 6	まとめ	158
第八章	多言語 InputMethod の IVS 対応	160
8. 1	はじめに	160
8. 2	IVS 対応に向けた環境の整理	160
8. 3	IVS 対応の実装	161
8. 3. 1	IVS 対応フォントの作成	161
8. 3. 2	IVS 用変換辞書作成	161
8. 3. 3	MySQL への登録	163
8. 4	IVS 対応の検証	165
8. 5	おわりに	167

第三部 結論	169
第九章 まとめおよび今後の課題と展望	170
9.1 はじめに	170
9.2 まとめ	170
9.3 今後の課題と展望	176
参考文献	180
謝辞	189
研究業績	190
1. 関連論文の印刷公表の方法及び時期	190
2. その他（研究発表：海外）	191
3. その他（研究発表：国内）	191
4. 表彰	193
著者略歴	194
1. 学歴	194
2. 職歴	194
3. 所属学会	194
付録	1
付録1 多言語文字入力支援システムの評価実験解答用紙および配布資料	1
アンケート	1
A-1 日本語	3
A-2 日本語	6
B-1 英語	9
B-2 英語	12
C-1 ヒンディー語	15
C-2 ヒンディー語	18
C-3 ヒンディー語（辞書表示あり）	21
D-1 ポルトガル語	24
D-2 ポルトガル語	27
【資料1】 Google 検索の方法	30

【資料 2】 AjaxIME 入力規則	33
【資料 3】 ソフトキーボードのキー配列	37
付録 2 JIS X 0208:1997 から JIS X 0213:2004 で例示字形が変更された字形リスト	1

図目次

図 1	我が国の IT 戦略の歩み	2
図 2	本研究の学術的背景との関係	5
図 3	カナ漢字変換の処理フロー	23
図 4	モジュールとしてのフォント	26
図 5	JIS X 0208 の包摂基準の例	31
図 6	日本語符号化文字集合の変遷	33
図 7	OSS と OSS ライセンスの関係	40
図 8	OSI 承認プロセス	81
図 9	異体字、外字の関係図	86
図 10	IVS,IVSC,IVD の関係	88
図 11	Adobe-Japan1 IVD における辻の例	89
図 12	Adobe-Japan1 IVD における籠の例	89
図 13	オープンな環境での外字の扱い方	91
図 14	行政システムへの提案	92
図 15	グリフデータベースの概要	96
図 16	Ideographic Description Characters	99
図 17	Safari4.0.1 での SVG フォント表示例	105
図 18	簡易検索	106
図 19	コード検索	107
図 20	部首検索（部首画数索引）	108
図 21	部首検索（部首名索引）	108
図 22	IDS 検索	109
図 23	IDS の例	109
図 24	IVS 検索例（1）	109
図 25	IVS 検索例（2）	110
図 26	IVS 検索例（3）	110
図 27	登録情報から検索	111

図 28	グリフ変更指示書	112
図 29	簡易検索 訓読み「から」で検索	115
図 30	部首検索 (部首画数索引) 「にすい」で検索した例.....	116
図 31	インターネット上で使用される言語のトップ 10.....	120
図 32	アラビア文字の字形変化	121
図 33	Unicode 3.2 で追記された脚文字を表す符号列	123
図 34	Operating System Market Share(2009,10).....	128
図 35	IME パッドでの文字一覧.....	129
図 36	The Unicode Standard Version 5.2.0 による Devanagari の追加 ..	130
図 37	Devanagari のレンダリングが出来ている例	131
図 38	Devanagari のレンダリングが出来ていない例.....	131
図 39	ASCII 領域内入力文字一覧	135
図 40	ヒンディー語のテーブル例	139
図 41	入力言語の選択	141
図 42	ヒンディー語入力例.....	142
図 43	検索エンジンへの遷移	143
図 44	クライアント・サーバー間の通信	144
図 45	入力実験方法.....	145
図 46	Virtual Keyboard v3.5.3	148
図 47	葛飾区や葛城市の例.....	162
図 48	IVS 用変換辞書	162
図 49	UTF-8 のビットパターン	163
図 50	多言語 InputMethod の DB 構造	164
図 51	IVS 用変換辞書の DB への登録.....	164
図 52	葛城市の入力 (IVSC なし)	165
図 53	葛城市の入力 (IVSC あり)	166
図 54	メモ帳での表示	166
図 55	日本語パブリックフォントの位置付け.....	178

表目次

表 1	言語と文字の関係（筆者作成）	17
表 2	日本における各フォント技術の出現	24
表 3	OpenType フォントのファイル構造	25
表 4	JIS X 0208 規格の推移	29
表 5	JIS X 0213 規格の推移	30
表 6	IPA フォントの利用形態と主な利用 OS	51
表 7	各国でのフォントおよびタイプフェイスの知的財産権の状況	66
表 8	フォントライセンスの許諾内容	72
表 9	OSD10 項目と IPA フォントライセンスの対応	79
表 10	文字コード情報	97
表 11	文字情報	98
表 12	登録情報	100
表 13	利用情報	100
表 14	グリフデータベースの試用結果	113
表 15	被験者実験の実験 No. と内容	150
表 16	正答数（対応のある t 検定（両側検定））	152
表 17	正答数（対応のある一元配置の分散分析）	152
表 18	誤答数（対応のある t 検定（両側検定））	153
表 19	誤答数（対応のある一元配置の分散分析）	154
表 20	回答率（対応のある t 検定（両側検定））	156
表 21	回答率（対応のある一元配置の分散分析）	156

第一章 序論

1. 1 本研究の社会的背景

1. 1. 1 日本語文字処理環境

コンピュータの出現、インターネットの普及とブロードバンド化によって、コンピュータ上で動くソフトウェアはもとより、オフィス文書、電子メール、書籍、画像、音楽等が、商用・非商用を問わずデジタル化され、インターネット上で交換されるようになった。デジタル化による技術革新は、社会の至る所に変化をもたらした。オフィス業務は IT 化が進み、個人においてもパソコンや携帯電話等でさまざまなデジタルデータの交換を行っている。

インターネット上の情報発信空間の確保は、企業が自社でコストを投入して整備するサーバ以外にも、動画共有サイトや SNS、Blog という CGM (Consumer Generated Media) と呼ばれるサービスによって個人にも容易なものとなった。さらに、オープンソース・ソフトウェア (以下 OSS) のようなソフトウェア開発手法の推進や、スキャナやデジタルカメラなどのデジタル機器とマルチメディア処理ソフトウェアのコモディティ化により、企業活動だけではなく個人が情報を作成し、発信する、誰もがクリエイターとなりうる時代となった。

これまで、情報通信基盤については、インターネットや携帯電話に代表される移動通信体の出現と発展・普及する中で、さまざまな機会にその整備の必要性が訴えられてきた。政府が掲げる情報通信政策である「e-Japan 戦略¹」に続く「u-Japan 政策²」では (図 1)、ブロードバンド基盤の全国的整備、有線・無線のシームレスなアクセス環境の整備というように物理的な意味での基盤整備が訴えられているし、ソフトウェア的な基盤についてはプライバシー保護、情報セキュリティの確保、電子商取引環境の整備など IT の利用環境整備へと政策課題が拡大しており (岡崎、

¹ 2001 年 1 月 22 日、IT 戦略本部は、e-Japan 戦略として IT 国家戦略を策定した。
e-Japan 戦略 (要旨) :

http://www.kantei.go.jp/jp/it/network/dai1/0122summary_j.htm

² 2004 年 7 月、内閣官房で主導する「e-Japan 戦略」の後継戦略として、総務省がユビキタスネット社会実現に向けた政策として u-Japan 政策を発表した。

u-Japan 政策 : http://www.soumu.go.jp/menu_seisaku/ict/u-japan/index.html

2004:118)、これらの整備に関わる研究が多数行われてきた。

我が国のIT戦略の歩み

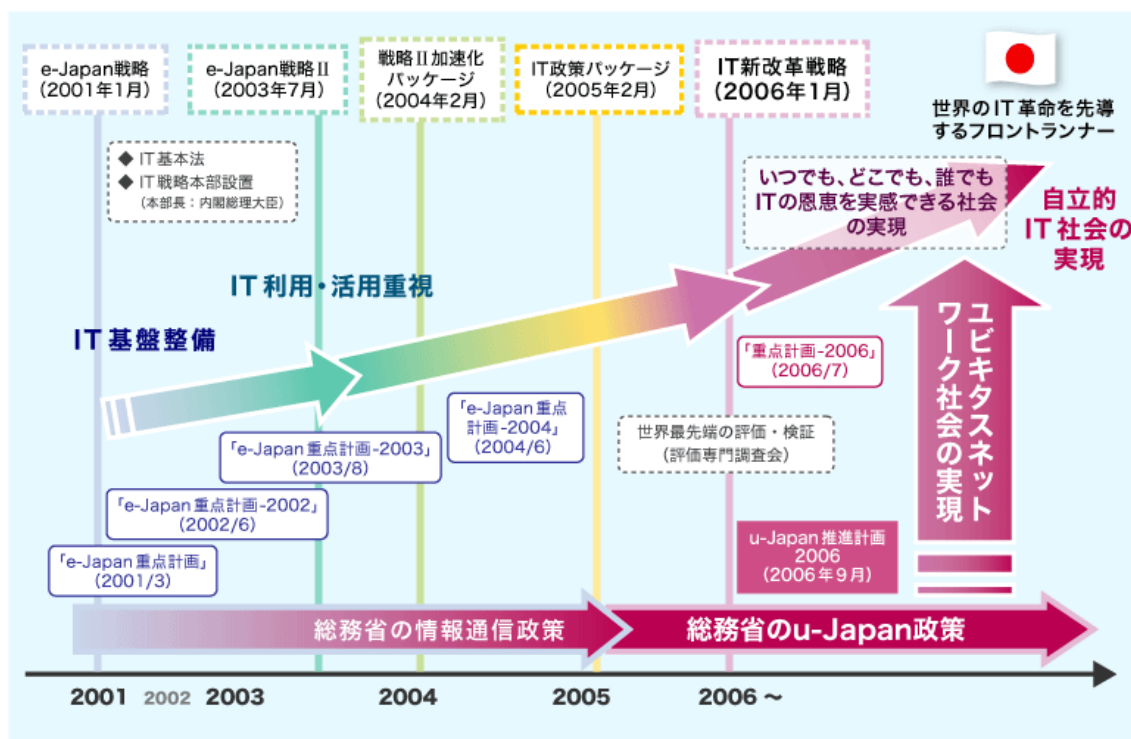


図 1 我が国の IT 戦略の歩み
(総務省 Web ページ³より引用)

さらに、2008 年には、我が国の国際競争力が低下している状況を踏まえ、我が国の国際競争力を強化する観点から、ICT 分野の研究開発・標準化に関する具体的な推進方策として、総務省により UNS 戦略プログラム II が提言されている。UNS 戦略プログラムは、u-Japan 政策を支えるとともに、ユビキタスネット社会に向けた社会の潮流を展望し、今後、重点的に推進すべき ICT 研究開発の方向性を、「国際競争力の維持・強化」、「安全・安心な社会の確立」、「知的活力の発現」とした上で、①国際社会を先導する「新世代ネットワーク技術戦略」、②安心・安全な社会を目指す「ICT 安心・安全技術戦略」、③知的創発を促進する「ユニバーサル・コミュニケーション技術戦略」を柱とする国家戦略である（情報通信審議会答申、2007:12）。

一方、我が国の情報通信基盤として、多くの ICT 研究において日本語の文字を入

³ 総務省「e-Japan 戦略」の今後の展開への貢献：
http://www.soumu.go.jp/menu_seisaku/ict/u-japan/new_outline01.html

かし表示するための日本語処理環境は欠かせない存在であり、その基盤があることを前提で推し進められている研究は多数存在する。しかし、コンピュータ上で日本語を扱うということは、1 万字以上の文字を入力し、それを文字という可視的な形として出力するための独自の文字処理環境が必要であるため、現在も、多くの課題を抱えている。

その一つとして、オープンな日本語フォントの不在という問題がある。インターネットの普及により国際化が叫ばれる中、日本の ICT 産業が競争力をもって開発を推進するため、日本政府は OSS の普及を推進してきた。しかし、OSS における開発環境、異なる OS 間におけるマルチ OS 環境で、日本語の処理、表示を行うためには日本語フォントは必須であるが、これまで OSS で利用可能な日本語フォントが存在していなかった。オープンな日本語フォントの不在が OSS 活動さらには日本のソフトウェア産業の活性化の大きな障壁となっている。

また、我が国の社会基盤として日本語の文字処理環境を見た場合、人名や地名あるいは古典などにおいて、異体字や外字といわれる漢字をどのようにネットワーク上で扱うか、という問題が存在する。電子政府、住民基本台帳等のデジタル化や、教科書等の教育用コンテンツのデジタル化にあたり、デジタル化がアイデンティティや文化の喪失を招くという印象さえ与えかねない現状に対し、異体字や外字への対応が期待されている。

1. 1. 2 多言語・多文字処理環境

ICT 化の進展と共に、世界中で多言語・多文字情報資源が蓄積され、インターネット上で発信される時代となった。コンピュータ上で扱うことのできる文字は、コンピュータ黎明期にはラテン文字が中心であり、ASCII における 7 ビット文字コードに代表されるように、少ない文字数しか扱うことができなかった。しかし、各国語対応の ICT 環境の進展と、インターネットの普及により、世界中の言語に用いるあらゆる文字を統一的に扱おうとする国際標準規格：ISO/IEC10646 Universal Multiple-Octet Coded Character Set⁴および Unicode⁵が規格化された。これによつ

⁴ JTC1/SC2/WG2 - ISO/IEC 10646 - UCS : <http://std.dkuug.dk/jtc1/sc2/wg2/>
日本の対応規格は JIS X 0221 (国際符号化文字集合)

⁵ The Unicode Consortium : <http://www.unicode.org/>

て、世界中で共通した文字コードにより、多言語・多文字情報資源を蓄積するベースが整備された。

しかしながら、これらの情報資源に対して実際にアクセスするなどの活用手段は、特定の言語および文字に限定されているのが現実である。ここには、言語、文字、知識、文化の「壁」が存在する。ユーザーにとって既知の言語や既知の文字が使用されている情報資源のみにアクセスが集中している。

総務省による UNS 戦略プログラム II の「ユニバーサル・コミュニケーション技術戦略」の中でも、「スーパー・コミュニケーション」として音声翻訳技術やテキスト翻訳技術といった多言語にわたる言語処理技術の研究開発が課題として取り上げられている（情報通信審議会答申、2007:104）ように、言語、文字、知識、文化の「壁」を超えるためのコミュニケーション技術が求められている。

1. 2 本研究の学術的背景

1. 2. 1 文字処理とは

本研究で論じる文字処理とは、異なる2つの研究分野を対象としている（図 2）。1つは、コンピュータ上で文字を扱うためのソフトウェアとしての情報工学における研究である。もう一つは、ネットワーク社会の中で文字とはどうあるべきか、あるいは文字情報を ICT の中でどのように有効利用するか、といった社会情報学における研究である。

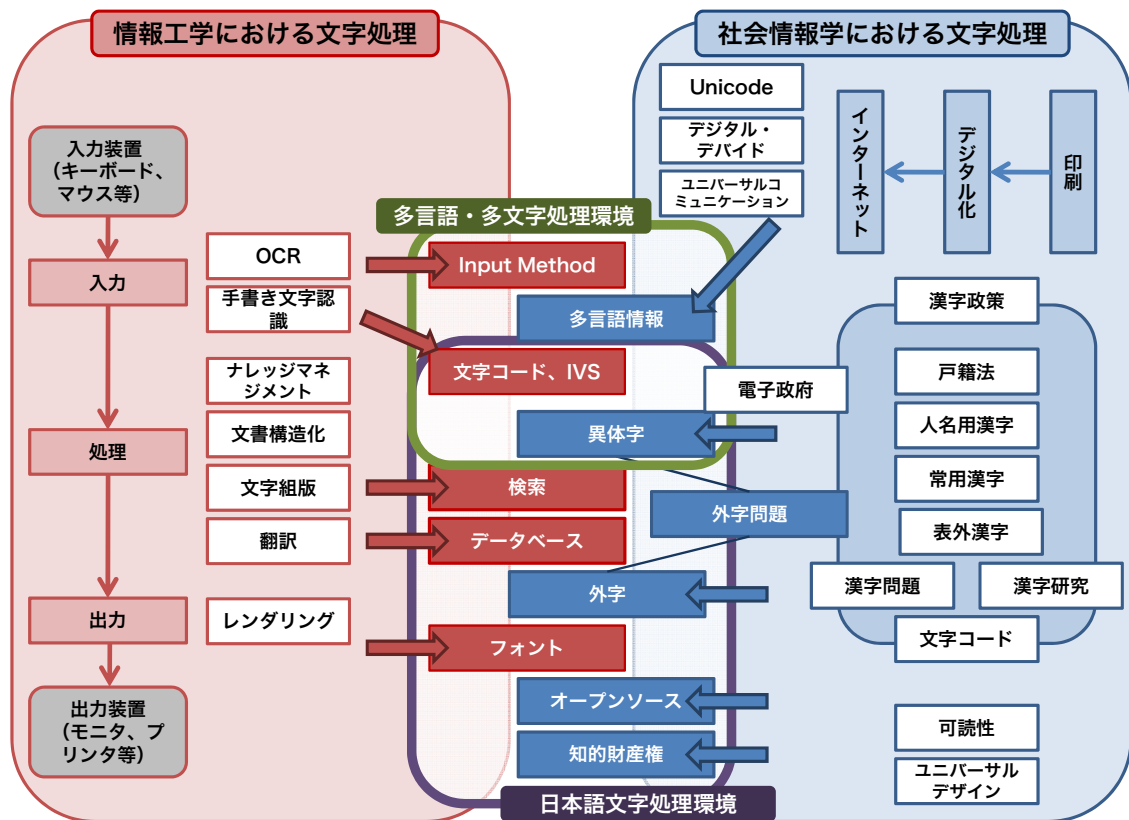


図 2 本研究の学術的背景との関係

(1) 情報工学における研究

コンピュータ上での文字処理を単純モデル化すると、入力、処理、出力に分けることができる。本研究では、以下のように定義する。文字処理上の入力とは、なんらかの方法で任意の文字をコンピュータ上で扱うための文字コード列に変換する技術を指す。処理とは、入力された文字コード列を利用用途に合わせて加工、編集する技術を指す。出力とは、文字を可視的な文字画像としてディスプレイ上あるいは印刷物として表示することを指す。

「入力」について研究の1つは、キーボード等の入力装置を利用した文字入力方法である。これまでの文字入力方法としては、ラテン文字やインド系文字、アラビア文字などにおいては、キーボードに文字コードが直接アサインされた直接入力方法がとられている。従って、所望の文字入力を行うためには、その文字に対応したキーボード配列を規定し、それを変更する手段が OS レベルで実装されている必要がある。

一方、日本語や中国語、韓国語などの多文字圏では、InputMethod というソフトウェアが必要となる。

日本語における InputMethod の技術は、森ら（1978）による「かな漢字変換」にはじまる。かな漢字変換は、入力したい文章の読みをキーボードから入力し、意味・文法解析を行い、読みと漢字との変換対応表（かな漢字変換辞書）を用いて、漢字仮名交じり文に変換する処理である。現在は、この技術を応用した InputMethod として、OS 標準搭載の MSIME（Windows）、ことえり（Mac OS）や(株)ジャストシステムの ATOK などがある。また、田中ら（2003）による携帯等の少数のキーでの入力の研究、高林ら(2002)、市村ら(2002)や佐藤ら(2006)による入力の変換効率の向上や予測変換機能などの研究が続けられている。さらに、清田ら（2007）による視覚障害者の日本語文字入力支援システムや、山口ら（2007）による視線移動を利用した肢体不自由者に応用可能な入力方法の研究も行われている。

画像として取り込んだ文字画像を文字列に変換する技術の研究として手書き文字認識技術や光学文字認識技術がある。文字認識の研究は、パターン認識の研究の中でも歴史が長く、1928 年には印刷数字の OCR (Optical Character Reader)の特許がオーストラリアで出願されている。1955 年には、アメリカのファーリントン社が計算機を使用した OCR を開発している。日本では、1996 年に通産省のパターン処理大型プロジェクトがスタートし、同年、東芝が、手書き数字認識を応用した郵便番号読取り装置を開発している。1984 年には電子技術総合研究所（現：独立行政法人産業技術総合研究所）が 3,036 字種(平仮名 71 字種+JIS 第一水準 2965 字種)、607,200 文字からなる手書き文字データベース ETL9B を公開し、大規模手書き文字認識の研究が盛んに行われるようになった。現在では、ETL9B の 3,036 字種に対し、99%以上の認識精度を誇るコンピュータによる手書き文字認識システムが開発されている（澤 他、2001）。また、TV 映像中のテロップや写真画像などから文字情報を抽出する研究も進められている。

手書き文字認識としては、前述のような紙の上に書かれた文書イメージを光学スキャンすることで認識する「オフライン手書き文字認識」と、特殊なタブレットや PDA において、ペンまたはスタイラスを用いて入力領域に文字を書き込む「オンライン文字認識」がある。

本研究の定義する情報工学における「処理」の研究としては、データベース、文

書構造化、文字組版、翻訳技術、あるいは文字情報を元にしたナレッジマネジメント、検索技術など多岐にわたる。

「出力」においては、文字を画像として表示するためにフォントとレンダリング技術が必要となる。フォントとは、文字コードに対応した文字図形データの集合である。日本語フォントについてはこれまで、田中ら（1995）や上地（2002）等によりフォントの自動作成技術の研究、また可読性を意識したフォントの制作も行われてきた。

レンダリング技術としては、文字のアンチエイリアシング方式が研究されており、WindowsOSでは、「サブピクセルレンダリング」である「ClearType」技術によりディスプレイのR, G, Bの各サブピクセルを発色させ、その色調を微妙に変化させることで、実解像度以上（横方向の解像度を3倍する）の繊細な文字表示を可能にしている。また、ClearTypeの根幹技術として、ヒンティングとスムージングがある。ヒンティングとは文字の見た目を美しくするため、文字を構成する線の太さ・幅を調整する技術のことである。スムージングはアンチエイリアスと同様、ピクセルのぎざぎざを微妙な発色の違いによって埋める技術のことである。ヒンティング情報はフォントに埋め込まれるため、この情報が多いフォントほど美しく表示できることになる。従ってフォントのクオリティに寄るところが大きく、表示品質を向上させるためのよりいっそうの研究開発が期待されている。

（２）社会情報学における研究

ネットワーク社会の文字処理においては、文字は文字コードに変換されたビット列として流通することになる。紙の上の手書き文字や印刷文字を媒体として情報を伝達していた時代には、可視的な文字がすでに存在し、それを読むことでコミュニケーションが図られたが、コンピュータの出現により、文字情報は、デジタルで扱える文字コードに変換され、文字コードはフォントにより可視化されるようになった。閉ざされたシステムの中では、文字コードは自由であっても問題とならないが、システムのネットワーク化が進み、さらにインターネットが普及したグローバル社会では、どの文字をどのビット列（文字コード）に対応させるかが情報交換において重要となる。

日本語の場合は、デジタルで取り扱う文字コード以前に、政策としての漢字問題

が存在する。戸籍法、人名用漢字、当用漢字から常用漢字に続く「一般の社会生活において現代の国語を書き表すための漢字使用の目安」が政策として示されている。これらは、常用平易な文字の使用を政府サイドが設定する文字の制限として認識され、その文字の範囲の定義やそもそも制限を設けることの反発など長年論議となっている。ここで定義される文字についても、社会的な変化により見直しが図られており、例えば常用漢字については、2005年に「情報化時代に対応する漢字政策の在り方を検討することが必要」であるとした報告書（文化審議会国語分科会、2005）により見直しが図られ、文化審議会国語分科会の漢字小委員会が2009年現在もなお審議を行っている。漢字政策における見直しには、「国語に対する世論調査」や漢字使用の頻度数調査、読み書き能力調査、人名・地名等の固有名詞調査などとともに生活実態と照らし合わせて日本の漢字をどのように考えていくかという議論が必要となるため、多くの有識者が議論に加わっている。漢字政策の見直しでもわかるように、コンピュータ化、ネットワーク化により、漢字が簡単に入力でき、コピー・ペーストにより大量の文字が利用出来るようになったため、手書きでは利用しなかったような漢字さえも常用平易な漢字へと格上げされる傾向となった。表外漢字字体表（国語審議会、2000）は、まさにワープロ等の普及により字体の混乱を防ぐためにその選択の拠り所として作成されたものである。また、当然コンピュータで利用可能な漢字とは、文字コードで定義されている文字であり、漢字政策と文字コードは、互いに影響を及ぼす関係にあるといえる。

これら、漢字政策については、氏原（2006）の常用漢字表に関連した漢字政策や、円満寺（2005）による人名用漢字の研究、複雑に入り組んだ文字コードと漢字問題については安岡ら（1999）の文字コード研究などがある。また、漢字については、どの文字をどういった形で表現するか、という問題がある。常用漢字表や文字コード表を作成（印刷）するにあたり、それを可視的に示すためには例示字形が必要となり、「いわゆる康熙字典体（直井、1999:184）」が基準として用いられているが、フォントを制作するに当たっては、この例示字形を参考にしつつ、フォントのデザインポリシーの統一のために漢字の成り立ちから知ることが必要になり、白川（1984）、（1987）、（1996）や阿辻（1994）等の漢字研究が重要となる。

また、デジタル化によりフォントが簡単に複製し改変することが可能となったため、フォントやタイプフェイスの知的財産権についての研究が丸山（2006）や中塚

(2008)らによって行われているが、フォントやタイプフェイスの制作者にとって満足の行く状況には至っていない。

こういった文字を使った情報の流通としては、電子政府・電子自治体の基盤形成があげられる。須藤(2004:132)によると、電子政府構想の主要な目的はITと情報ネットワークを基盤にした「市民を中心にした政府」の構築である。インターネットを使った電子申請や公文書交換にあたって住所、氏名等に使う文字は欠かすことができない。汎用電子情報交換環境整備プログラム(石崎、2006)(関口、2006)では、電子政府の基盤となる文字情報の整理・体系化が試みられている。しかしながら、電子政府によるワンストップサービス⁶の実現にはまだ十分な環境が整っていない状況にあり、その推進が期待されている。

一方、ユニバーサル・コミュニケーションの1つとして、言語グリッド(石田、2008)のような多言語サービス基盤の研究が進められており、一連の研究の中で、文字処理研究としてはCHOら(2006)が絵文字コミュニケーションの検討を行っている。

国際情報通信政策の1つとして、経済的発展および民主化を容易にする情報通信技術へのアクセスを、全世界の人々に適当な価格で可能とする(デジタル・デバイドの解消)のための発展途上国にたいする国際協力は不可欠(岡崎、2004:119)とされる。三上(2000)はデジタル・デバイドの観点から調査・研究を進め、デジタル技術の当該言語へのローカリゼーションの進展がデジタル・デバイド克服のプロセスであると述べている。デジタル・デバイドの解消を含め、言語の壁をこえるための研究が期待されている。

このほかに、文字認知に関わる研究として、種村ら(2006)による携帯電話等の電子ディスプレイでの文字の黒みによる可読性の評価や、工藤・成田(2005)のハイビジョン番組の中の字幕呈示パラメータに関する研究などの文字の可読性に関わる研究や、鷲巢(2009)によるユニバーサルデザインの視点での文字のデザイン研究

⁶ 政府による「行政情報化推進基本計画(1994年12月25日閣議決定、1997年12月20日改定)」によると、ワンストップサービスとは国民生活、企業活動等に必要行政手続、行政情報の提供等について、地方公共団体等との連携・協力を図りつつ、情報通信技術を活用した手続の案内・教示、必要な行政情報の提供、各種施設の利用案内・予約、申請・届出等の受付、結果の交付等の行政サービスを総合的・複合的に提供するサービスをいう。

が行われている。

1. 2. 2 日本語文字処理環境

本研究における日本語文字処理環境の検討は、それを情報通信基盤として十分に機能させるために近々に整備すべき部分をターゲットにしている。

1つは、文字処理上の出力に必要であり、かつ漢字政策や文字コード問題に影響を受けるフォントである。フォントは、IT コミュニケーションを支える情報通信基盤として重要なものの一つである。工学的研究の項で示したように入力、処理過程を経て文字を可視的に表示するためには、フォントは必須である。日本語フォントについてはこれまで、田中ら（1995）や上地（2002）等のフォントの自動作成技術や、安岡ら（1999）等の文字コード研究、守岡ら（2004）等の漢字情報データベースに関する研究など、個別技術に関する研究は盛んにおこなわれてきた。しかしフォントを情報通信基盤として明確に位置づけ、それを基盤として整備するためにどのような条件が必要か、どのような整備の方針がありうるかを検討した研究はなく、本研究はこれを試みるものである。

フォントを情報通信基盤として整備するうえでの重要な着眼点は、モジュール化・オープン化である。Carliss Y. Baldwin, Kim B. Clark（2000=2004）はパーソナルコンピュータにおいてモジュール化の発想がいかに重要であり、それがいかにイノベーションを創出し、ソフトウェア産業、シリコンバレー、ネットワーク経済を生んだかを示した。国領（2003:72）は多数の技術の複合したシステムにおける自律・分散・協調の基盤を支える重要な設計思想をオープン（開かれた構造）化とした。オープン化の前提としてモジュール構造がある。

しかし、モジュール化はイノベーションを加速させる側面を持つが、モジュールは共通のインターフェースを持つため冗長性を内包しており（青島・武石、2001:43）、システムが最適なパフォーマンスを得るためには構成要素間の情報を仲介する機能が重要となる。

日本政府が OSS を推進する理由には、ソフトウェア技術の中での知識共有によるイノベーションの促進があげられる（田代、2006:540）。社会構造の変化の中で、日本経済の成長・発展を促すためのイノベーションの促進を支えるために求められているオープンな日本語フォントを定義することは、情報通信基盤の整備と言える。

日本語フォントも、パーソナルコンピュータの出現とネットワーク社会化により、モジュール化へと進み、クローズドなモジュールからオープンなモジュールへと変化していったと考えられる。そして、日本語フォントのモジュールとしての位置付けを明確にし、日本語フォントが最適に機能するための情報を仲介するシステムをデザインすることが、日本のソフトウェア産業のイノベーション促進へとつなげるための課題である。

さらに行政システムの IT 化にあたって、従来から問題となっていた外字問題にたいして、情報交換性や相互運用性といった観点からの提案が必要とされている。

1. 2. 3 多言語・多文字処理環境

日本語という 1 つの言語の文字処理環境をとらえた場合でさえ多くの課題が存在するため、さらに多言語・多文字に対象を広げると、より多くの文字処理環境の整備に関する課題が存在することになるが、本研究では、すでに Web 上に存在する多言語・多文字情報資源へのアクセスを可能とすることを目的とする。

多言語情報資源へのアクセスのための第一歩は、情報資源に含まれる文字列を入力し、それを検索クエリーとして送信する手段を持つことである。したがって、多言語 InputMethod が、多言語・多文字処理環境の第一歩といえる。

これまでラテン文字やインド系文字、アラビア文字などにおいて文字入力、キーボードに文字コードが直接アサインされた直接入力方式がとられている。従って、所望の文字入力を行うためには、その文字に対応したキーボード配列を規定し、それを変更する手段を OS レベルで実装されている必要がある。

一方、日本語や中国語、韓国語などの多文字圏では、InputMethod というソフトウェアが必要となる。

これらの、直接入力方式または InputMethod による変換入力方式のいずれも、どの言語（スクリプト）を入力できるかは、使用 OS 環境の設定に依るところとなる。

しかし、多くのユーザーにとって、ラテン文字と母語以外の文字を入力する環境を整備し、実際に文字入力をするのは非常に難しい状況にある。さらに、国際化する中、海外のインターネットカフェ等のオープンな環境で、Web 上の検索を行う際にも、検索するための文字が入力できない、という言語の障壁が立ちほだかる。

また、多文字の一例として日本語がある。これまで符号化文字における日本語入力については、多数の研究が行われ一定の水準に達しているが、異体字の処理方法については、文字入力を初めとしてその処理方法の提案が期待されている。

1. 3 本研究の目的と方法

本研究の目的は、我が国の情報通信基盤として、日本語および多言語の文字処理環境を整備するために、いくつかの必要な貢献をすることである。

第一部では、日本語文字処理環境の研究を行う。

まず、第二章では、日本語文字処理環境の概括と問題の所在を明らかにする。

その上で、第三章において、情報通信基盤として必要とされてきた日本語パブリックフォントについて、技術と社会環境の変化により新たなドメインが必要とされてきた状況を整理し、本研究で実施した IPA フォントでのオープンフォントの整備事例を提示し考察することで、今後の日本語パブリックフォントの課題を提示する。

方法としては、日本語フォントにとって OSS が大きな社会変化であると捉え、いままで未整理であった日本の OSS 政策とその中での日本語フォントの位置づけを整理分析する。さらに、海外における国家レベルでの OSS とフォント政策を見ることで、社会環境の違いがフォント技術にもたらしている影響を検証する。そして、日本の OSS 政策の中で公開された IPA フォントを取り上げ、現在までの経緯と問題を整理・観察した上で、公共フォントとしての日本語フォントの課題を提示する。

第四章では、第三章で明らかにした情報通信基盤を支える日本語パブリックフォントのライセンスの研究を行う。フォントは、その性格上プログラムとコンテンツの両方の要素を含んでいるが、加えてフォント特有の要素もあり、かつ日本語フォントには日本語特有の保護すべき要件を備えている。本研究では、情報通信基盤を支えるデジタル財の1つであるオープンな「日本語パブリックフォント」に最適な新しいライセンスを、デジタル著作権表明 (Digital Rights Expression : DRE) の視点に立脚して検討する。

方法としては、まず、情報通信基盤を支える「日本語パブリックフォント」のライセンスを検討する上での定義付けと要件を明確にする。同時に、GPL ライセンス (GNU General Public License) やクリエイティブ・コモンズなどを分析し、新たに日本語パブリックフォントに適したオープンソースライセンスを検討する。

第五章では、情報通信基盤としての漢字問題への1つの対応としてグリフデータベースを研究する。

方法としては、日本の社会基盤として既存のフォントでは解決できない漢字についての問題を明らかにする。その上で、漢字を文字図形として扱うグリフデータベースを定義し、グリフデータベースからの漢字の検索方法や検索したグリフを社会基盤の中でどのように利用するかについての提案を行う。

第二部では、多言語・多文字処理環境の研究を行う。

第六章では、多言語・多文字環境の概括と問題の所在を明らかにする。

第七章では、利用したい言語に不慣れな言語学習途上のユーザーでも簡単に所望の言語の情報資源にアクセスできるような、多言語に対応した入力支援システムの研究を行う。

方法としては、多言語に対応した入力支援システムの要件を整理し、その為のシステム提案を行う。具体的には、Unicodeで定義されているすべての言語を対象とするために、OSや対象言語の制約を持たない文字入力手段の提供として、ASCII領域内の文字を利用した変換辞書を検討する。また、ユーザーが簡単に、正確に所望の文字を入力するために、インクリメンタルサーチと、入力文字候補の関連情報を表示する機能を設ける。そして、実現手段として、インターネットブラウザで入力操作を行えるように Ajax (Asynchronous JavaScript + XML) 技術を用いたシステム実装を行う。実装したシステムについては、被験者による評価実験を行う。これにより、提案システムの有効性を評価するとともに、改善ポイントを抽出する。

第八章では、第七章で構築したシステムを利用して、日本語の多文字に対応した入力環境の実証として、Unicodeで規格化された Ideographic Variation Sequence (IVS) 対応による異体字表示の実装を行う。

方法としては、IVSに関する技術情報を元に、実際にフォントおよび多言語に対応した入力支援システムに異体字を実装し、IVS 解釈可能な環境を構築して、異体字の入力と表示が可能であることを検証する。

本研究の第一の貢献は、グローバル社会において日本だけでなく世界レベルで競争力を持つためにも必要とされる情報通信基盤としての日本語文字処理環境の課題を整理したことにある。

第二の貢献は、相互運用性に富んだ情報通信基盤の確立へ寄与する日本語パブリックフォントの定義付けを行ったことにある。

第三の貢献は、オリジナルフォントが保証する文字への信頼性確保の要件を明確にし、日本語フォントと情報通信基盤との関係性を明らかにした点にある。

第四の貢献は、文字への信頼性確保の要件と、OSI による OSD のオープンソース定義とを両立させて DRE の視点に立脚したライセンスとして日本で始めて具体的な要件を構築し OSI 承認を得たことにある。結果として日本語フォントをデジタル財として自由に流通できる環境を整えることができた。

第五の貢献は、行政システムにおける外字問題に対し、情報交換性や相互運用性を考慮した上で運用方針を示し、実際に利用可能なモデルを構築したことである、

第六の貢献は、我が国の情報通信基盤としての多言語文字処理環境の第一の課題を、多言語情報にアクセスするための手段であると捉え、多言語情報にアクセスするための OS や対象言語の制約を持たない文字入力手段の提供を行ったところにある。

第七の貢献は、IVS の実装を実際に行うことで、多文字環境の新たなサービスモデルの提案をしたことにある。

第一部 基盤としての日本語文字処理環境

第二章 日本語文字処理環境の概括と問題の所在

2. 1 はじめに

本章では、これまでの ICT における日本語文字環境を概括し、グローバル社会において日本だけでなく世界レベルで競争力を持つためにも必要とされる情報通信基盤としての日本語環境の課題について整理する。

2. 2 世界における日本語

日本語は単一の国で利用されている非常に珍しい言語である。井上(2001:78)によると、日本語には「三位一体説」が成り立つとされ、「言語」の使われる地理的範囲と、国家の範囲と、日本「民族」の住む範囲が一致しており、アイルランドを除けば世界で稀な言語である。そして、その表記に利用される文字についても、日本独自の発展をしている。

言語とは、人間の社会集団における相互伝達の手段としての本来音声による記号である。そして、言語は文字によって表現される。言語は、音と意味が恣意的に結びつけられた言語記号を単位とする体系でその規則は社会的慣習として存在する。各言語は独自の音韻体系、文法構造、語彙を持つ。現在、世界の言語は 6000 以上になると推定されるが、一方で言語の消滅していく速度は加速化され、5 割から 9 割が今後 100 年のうちに消滅すると言われている。理由は、グローバリゼーション、つまり地球規模の文化の均一化にあるといわれている(民族の世界地図、2000:47)。

文字とは、音と意味が結合して特定の言語を表す記号である。一字が一語を表す表意文字と、音だけに関連つける表音文字がある。さらに、文字が表す単位が単語・音節・音素という階層のどこに位置するかという基準によって、表語文字、音節文字、アルファベットに分類することもできる。表音文字は音節文字とアルファベットとに分かれ、表意文字のうち意味を表す最小の単位が語であるものは表語文字に対応する(三上、2005:921)。

世界の文字は、インド系文字、アラビア文字、ラテン文字、漢字等に分類される。そして、多くの言語は文字によって表現されるものであるが、その関係は単純に 1 対 1 の関係にはない(表 1)。

表 1 言語と文字の関係（筆者作成）

文字	インド系文字				アラビア文字			ラテン文字			漢字		ひらがな カタカナ
	デーヴァナーガリー文字	タイ文字	..	タイ語	アラビア語	ペルシャ語	ウルドゥー語	英語	フランス語	ドイツ語	中国語	日本語	日本語
言語	ヒンディー語	ネパール語	..	タイ語	アラビア語	ペルシャ語	ウルドゥー語	英語	フランス語	ドイツ語	中国語	日本語	日本語

例えば、ラテン文字は英語、フランス語、ドイツ語などの言語表記に使用される。このうち英語は、現在、世界の事実上の標準語といえるが、英語を母国語とする国は、英国、米国をはじめ、世界各国に多数存在する。一方、「公用語」という国レベルで公式に使用することを定め実務処理を円滑化するための言語があるが、例えばスイスでは、ドイツ語・フランス語・イタリア語・ロマンシュ語の 4 言語であり、使用する文字はラテン文字である。インドでは、公用語のヒンディー語と準公用語の英語の他に、アッサム語、ベンガル語、タミル語などの 17 の憲法公認語があり、それぞれが地方公用語として使われている、それぞれの言語はインド系のそれぞれの文字で記述されるため、インド一国内で多言語・多文字が必要となる（民族の世界地図、2000:49）。

このように国と言語と文字は、単純な対応関係にあるわけではない。

国と言語と文字は歴史とともに変遷しており、それは人の移動とともに広がり、変化、分岐あるいは消滅を繰り返してきた（東京外国語大学、2005）。人を介した自然な伝播だけでなく、国家の成熟とともに、使用する言語、文字について政治的に統制を図った例も少なくない。近年、例えばトルコでは、1928年にトルコ共和国初代大統領 ケマル・アタチュルクによってそれまでトルコ語を表記していたアラビア文字を廃止し、ラテン文字を採用した（野田、1999）。中国では、少数民族政策として、「国家通用言語文字法」の第八条に「各民族はいずれも自己の言語・文字を使用し発展させる自由を有する。少数民族の言語・文字の使用は、憲法と民族区域自治法およびその他の法律の関連規定に従うものとする。」とあり、政府として少数民族の言語・文字を保護することを明言しており、自治区毎で、新聞・出版や文書、市街表記、言語教育において独自の文字が用いられ、漢語（漢字）との併記を行っている。

こうした世界からみると、日本語は日本という島国によって閉じられた言語であり、日本語の文字環境を真剣に考えなくてはならない民族は日本人しかいない、ということができよう。

日本語は表記に、表意文字である漢字、表音文字であるひらがな、カタカナ、さらにはラテン文字や記号等を使用する世界でも稀な言語である。山口（2006）によると、日本語の文章を書き始めたのは645年の大化の改新以降とされ、当時は漢式和文で記述された。異なる言語体系⁷である中国から借りてきた漢字を利用して日本語を書き表す中で、日本語をうまく書き表せないというもどかしさから、漢字を表音文字として捉える万葉仮名が生まれ、やがてひらがな、カタカナへと展開していく。そして現代において、漢字仮名交じり文である日本語をICTの中で利用するためには、日本語ならではの工夫を必要としている。以下に、その過程として、文字情報の記録・伝達テクノロジーである印刷技術の日本語化の進展を整理する。

2. 3 印刷技術の中の日本語

紀元前2世紀に中国で紙が作られ、これはキリスト教ヨーロッパ諸国へと伝播する。中国ではさらに7,8世紀に木版印刷がはじまり、11世紀には活字を生み出している（H.G.ウェルズ、1966）。

日本には、764年に印刷された百万塔陀羅尼が、印刷された年代が明確な世界最古の印刷物として存在している⁸。その印刷方法については、版木に経文を彫って印刷した木版説と、銅板に文字を鑄造して印刷した銅凸版説の二説があり、そのどちらであるかは現在にいたってもわかっていない（印刷図書館、2009）。

15世紀にドイツ人ヨハネス・グーテンベルグが鉛製の活字とそれを組み合わせて印刷する活版印刷術を発明することで、書籍の増大とともに知的生活が活性化された（H.G.ウェルズ、1966）。

⁷ 日本語は、多くの助詞・助動詞が、実質的な意味を持つ単語に膠で接着したようにくっつく文法的な役割をもつことから「膠着語」と呼ばれる言語である。中国語は、日本語の助詞・助動詞に該当するものが少なく、文法的な役割は実質的な意味を持つ単語の順番で表すため「孤立語」と呼ばれる言語である（山口、2006:18）。

⁸ 国立国会図書館や印刷図書館に貴重書として所蔵されている。

国立国会図書館：<http://www.ndl.go.jp/exhibit/50/html/catalog/c002.html>

印刷図書館：http://www.printing-museum.org/collection/looking/02_01.html

日本に活版印刷術が伝わったのは 1590 年で、天正遣欧使節が印刷機や印刷機材を携えて帰国し、翌年には印刷にかかっている。これらは、「きりしたん版」と呼ばれるものであるが、1614 年のきりしたん追放令によって西歐式の印刷活動は終息し、次に西歐印刷術が日本に入るのは 250 年後の 1869 年である(小宮山、2009:25)

江戸時代初頭には「嵯峨本⁹」に代表される木活字印刷が普及したが、版本の主流は、活字ではなく従来の版木による整版印刷の本であった。これは、近代の活版印刷と異なり、組み直しに時間と手間がかかり、増刷のたびに校正を伴うなど、利便性とコストにおいて、劣勢であったことに起因する。量産を前提とした工業製品として欧米の活字製作技術が日本に導入されたのが 1869 年(小宮山、2008a:116) ということになる。

(1) 金属活字

金属活字は、木版のようにある媒体ごとに使い捨てるのではなく、再利用できることを前提に開発された。金属活字は、鉛を主体とした錫・アンチモンの活字合金で鑄造した四角い長方形の上面に、逆向きで文字や記号は凸の状態で浮き出ている。(小宮山、2009:26)。この金属活字の製法を確立したのが、ドイツのマインツの都市貴族であり金細工師のヨハネス・ゲーテンベルクである(Helmut Presser、1973)。

書体、サイズごとに文字セットが必要であり、紙面デザイン、レイアウトの自由度が低い、凸版印刷ならではの力強い印字が可能である。

日本語活字が本格化した時期は幕末以降である。本木昌造が、明朝体活字で印刷を行う上海の美華書館で電胎母型法を開発したウィリアム・ギャンブルを招聘し、活字鑄造法と印刷術を教わることで、のちに築地活版製造所と大蔵省印刷局の基礎を作る事となった(小宮山、2005)。

しかし、金属活字は写真植字の登場とともに衰退し、現在はほとんど金属活字を使用した出版物は刊行されていないが、いまだに出版物の文字を読むことを「活字を読む」、印刷物にすることを「活字にする」という様な表現が使われている。活字は中国語であり、英語では Movable Type という。活字の性格が、繰り返して使うことができ、どこで使われてもいつも同じ字形であることが基本であるならば(小宮

⁹ 角倉素庵と本阿弥光悦のよって木活字を使って印刷された刊本。刊行地の名前をとって嵯峨本という(小宮山、2009:206)。

山、2009:26)、現在の技術による印刷物に「活字」という言葉を使うことも間違いではない。

(2) 写真植字

「写真植字」(以下、写植)は、名前の通り写真技術を使って植字する方法であり、写植機を用いて文字などを印画紙やフィルムに印字して、写真製版用の版下などを作ることを言う。写植機は、ガラスの文字盤に光を当て、レンズによって拡大縮小して文字を印画紙やフィルムに露光する。永原(2008)によると、1920年頃のイギリスで、すでに写真を応用した文字組版の研究が進められていたが、実用にたつた写植機は日本で生まれ、世界に広がっていった。

写植機は、石井茂吉と森澤信夫という二人の日本人によって実用化され、1925年に特許が成立し、1929年に世界で初めて実用機が完成した。開発にあたっては、石井が写植文字の文字盤を、森澤が装置部分の開発を主に担当した。この二人は後に袂を分かち、それぞれ写研とモリサワという、写植時代をリードする2大企業を設立することになる。

永原(2008)によると、日本よりずっと機械文明が進んでいるイギリスで開発が成功しなかった理由を、アルファベットの活字が文字固有のプロポーションに添った幅で作られていたことが開発を妨げていたためと指摘している。つまり、欧文では一文字打つたびにその字に合った幅で印画紙を移動さなくてはならず、一方、日本の活字はボディが正方形であったため、一文字ごとに同じ幅だけ印画紙を移動させれば良いことから、開発を容易にし、新しい技術を生むこととなった。写植技術は、日本特有の多文字と日本語の活字組版の特徴が、技術イノベーションを牽引した事例といえよう。

さらに、手動写植機をコンピュータ化したシステムが電算写植機である。電算写植は手動の写植で困難だった文字の修正等を容易にし、それまでのワークフローを大幅に改善する技術であった。しかし、Macintosh OSやWindows OSの出現によりDesk Top Publishing(DTP)が可能となることで、電算写植はDTPに置き換わって行った。

2. 4 コンピュータの中の日本語

ICT の基盤として、コンピュータの出現と発展を除いて語ることはできない。“computer” が「算術計算を行う人を指す言葉」を意味していた通り、コンピュータが世界的に広まって行く過程で、1950 年～60 年代にかけて利用される分野の多くは数値処理であった。この処理に必要なとされる文字はラテン文字と数字だけで充分であり、100 文字程度の文字を扱うことができれば良かった。

コンピュータは 0 と 1 のビット列を処理する機械であることから、文字を処理するためには、1 つの文字に 0 と 1 のビット列からなる番号である文字コードを与える必要がある。コンピュータが異なる機器間で情報交換を行わない場合は、文字コードは独自の体系でよいが、情報交換を行うためには、統一したコード配列を必要とした。米国が情報交換用に定めた ASCII コード (American Standard Code for Information Interchange) は、7 ビットの文字コードであり、128 文字を表すことができる最も基本的な文字コード体系として、現在もなお利用されている。

当然、文字コードだけでは、可視的に文字を見ることができない。そこで必要となるのが、「フォント」である。コンピュータ黎明期には、文字コードに対応した文字図形データを 2 値画像として保持した「ビットマップフォント」により、モニタや紙への文字出力を行った。

また、文字の入力は、キーボードを介して行われるが、ASCII コードの文字はすべてキーボード上に配置することができ、従ってキーと文字コードは 1 対 1 の対応関係をとることができる。

一方、ASCII コードだけでは対応ができない日本語については、カタカナの入ができるように、ASCII コードを拡張した 8 ビットコードを用い、カタカナ入力可能なキーボードを開発することで独自の対応を行った。

コンピュータが、次第にさまざまな言語のドキュメントを処理するシステムへと展開していく中、日本では、1978 年に東芝が初めての日本語ワードプロセッサを商品化した¹⁰。世界初のワードプロセッサは 1964 年の IBM 社の Magnetic Tape Selectric 機であるが (Utterback、1998)、日本の場合、多数の文字の処理をしなくてはならないため、開発はたいへん困難なものであった。

¹⁰ コンピュータ博物館 日本語ワードプロセッサ年表
<http://museum.ipsj.or.jp/computer/word/index.html>

日本語処理の展開における大きな障害とは、日本語の文章の入力方法と、文字コードおよびフォントである。日本語の文章を入力するのは、漢字仮名交じり文を入力処理できなければならない。また、扱う文字については、どのような漢字を収納するのが最大の課題である。そこで、日本語の文字をコンピュータ上で扱うための文字コード体系と、それに合わせたフォントが必要となった。

ICT における日本語環境のもっとも基盤となる問題は、日本語の文字をいかに ICT 技術の中で実現するか、ということにかかっている。そして、社会的ニーズと同時に進展する ICT 技術の上に日本語環境の基盤を築くとは、文字入力環境と文字コードとフォントを整備するという他にない。

2. 5 日本語文字入力環境

日本語の文章を入力するためには、漢字仮名交じり文を入力処理できなければならない。

コンピュータ上でラテン文字系の言語を入力するには、利用する文字が 100 文字以下であるため、タイプライターと同様に、コンピュータ用のキーボードと文字コードが 1 対 1 に対応したシステムが設計されたが、漢字、ひらがな、カタカナ、英数字等の多数の文字を扱う日本語を入力するためには新たな文字入力方法を必要とした。

これまでの和文タイプライターでは、広い板の上の一つ一つ配置された漢字を選択すると言う方法がとられていたが、入力には熟練を必要とした。日本語文字入力装置としては、多段シフト方式、タブレット方式、邦文タイプ方式、2 打鍵文字コード方式、カナ漢字変換方式などが研究開発されてきた（森：1979）。

このうち、日本語ワードプロセッサの開発において出現した、日本語の InputMethod の基礎とも言える技術が「カナ漢字変換」である。森（1978:383）によると、カナ漢字変換は、入力したい文章の読みをキーボードから入力し、①文節、単語へのカナ文の分割、②辞書との照合③文法解析、④頻度学習機能、⑤同音異義語の選択などの機能から構成された処理を行う。キーボードを用いる場合の読みの入力方式には、カナ入力とローマ字入力があり、日本語対応キーボードとはカナ入力とローマ字入力に対応したキーボードということになる。

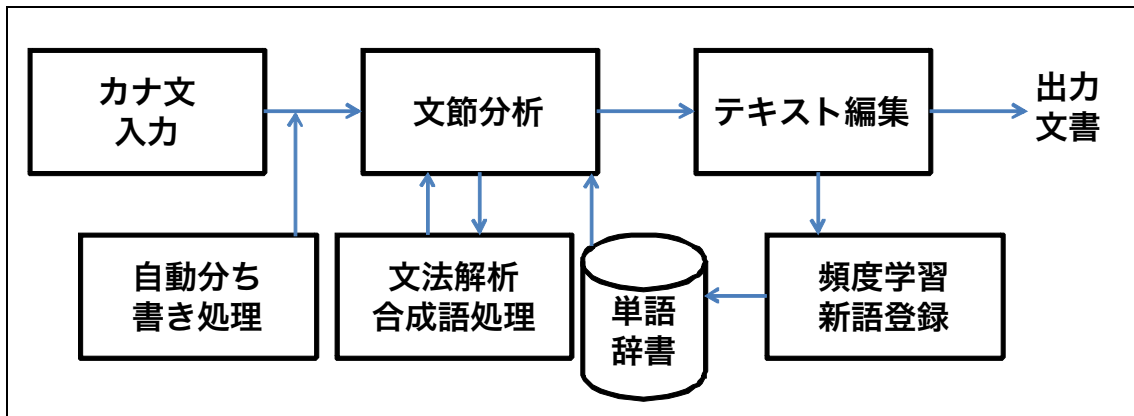


図 3 カナ漢字変換の処理フロー
(森 (1978)より筆者再作成)

2. 6 日本語フォント

情報を時間や空間を超えて伝達するためのメディアが「文字」であり、ある統一したデザインの文字群をコンピュータ上で再現するためにデジタル化した文字集合が「フォント」である。フォントとは英単語の「font」または「fount」のことで、金属活字時代にはあるデザインで統一された1つのサイズで統一された活字ひと揃いという意味であったが、現在ではコンピュータ上で利用するためのデジタルフォントのことを指すことが一般的となっている（小林、2005:23）。

デジタル化されたフォントには、文字図形データの持ち方によって2値画像の「ビットマップフォント」と、文字の形状を基準となる点の座標と輪郭線の集まりで表現する「アウトラインフォント」がある。

2. 6. 1 フォントフォーマットの変遷

「ビットマップフォント」は文字の形をコンピュータ上で表現するためのデータ形式の一つである。文字を小さな正方形の点(ドット)の集合として表し、ある決まった数の格子ですべての文字の形状を表現する。高速に処理することができるため、初期のコンピュータやプリンタはほとんどがビットマップフォントを使っていた。しかし、ビットマップフォントは拡大・縮小するとすぐに文字の形が崩れてしまうため、近年では、いくら変形しても文字の形が崩れないアウトラインフォントが主流となっている。ただし、アウトラインフォントはその処理に演算リソースを必要

とするため、多くのリソースを使うことができないメディア（例えば携帯電話）や特定サイズのフォントしか必要としない家電等の液晶ディスプレイ等には、ビットマップフォントが現在なお使用されている。

「アウトラインフォント」は、文字の形状を、基準となる点の座標と輪郭線の集まりとして表現する形式である。表示・印刷時に曲線の方程式を計算して、描画する点の配置を決定するため、拡大や縮小、変形をいくらおこなっても文字の形が崩れないのが特長である。ただし、ビットマップフォントに比べて処理に時間がかかるため、普及し始めたのはコンピュータの性能が飛躍的に向上した最近のことである。

アウトラインデータとしては、初期段階ではベクトル情報で輪郭を現すベクターフォントを使用していたが、現在は多次元曲線で表されており、2次スプライン曲線方式の TrueType フォント（Apple Computer 社と Microsoft 社の共同開発）とベジェ曲線方式の PostScript フォント（Adobe Systems 社の開発）がある。

PostScript フォントはさらに内部構造の違いのある OCF フォント（Original Composite Font）、CID フォント（Character Identifier Font）、OTF フォント（OpenType Font）に分かれる。表 2 は、日本における各フォント技術の出現経緯である。

表 2 日本における各フォント技術の出現

年	フォント技術
1986 年	和文ビットマップフォント登場
1986 年	和文 OCF フォント登場
1992 年	和文 TrueType フォント登場
1998 年	和文 CID フォント登場
2002 年	和文 OpenType フォント登場

一般にフォントは、

(1) タイプフェイスのデザイン（字形の輪郭線）を高次の数式で「表現（描画）したプログラム」

(2) 文字組を行う際の検索文字コードデータ・字巾データ・縦横組切り替え字

種データ・異体字データなどの「文字属性データベース」

から構成されており、プログラム及びデータベースの著作権で保護されている（日本タイポグラフィ協会、2003）とされている。

フォントフォーマットとは、(1)(2)の記述方法と構造を定義したものである。

現在は、TrueType フォントの後継フォーマットとして、Adobe Systems 社と Microsoft 社が共同で設計した OpenType フォントフォーマットが公開されている。OpenType フォントは、TrueType フォントの発展形であり、PostScript 形式のアウトラインも内包できるようにしたフォントフォーマットであり、マルチプラットフォームで活用が可能である（Microsoft、2001）。また、広範な言語への対応と多数の文字属性を記述する表 3 のような Feature テーブルを持っており（Microsoft、2008）、事実上 OpenType フォントがデファクトスタンダードとなった。

表 3 OpenType フォントのファイル構造

テーブル名	内容の概要
head	フォントヘッダー
name	フォントの名前、Copyright などのデータ
glyf	グリフデータ（字形のアウトラインデータ）
cmap	グリフへのコードマッピングデータ
GSUB	グリフの置換データ（縦横切り替えなど）
・・・など	・・・

2. 6. 2 フォントのモジュール化

このフォント技術の変遷を、パーソナルコンピュータを前提としたアーキテクチャの中で捉えると、ネットワーク社会の中で、データでの情報交換を可能とする環境への対応として、標準化・規格化の必要に迫られ、結果として、フォントはクローズドなモジュールからオープンなモジュールへと変化を遂げたとみることができる。

国領（1999:21）は、オープン・アーキテクチャ戦略を「本来複雑な機能を持つ製品やビジネスプロセスを、ある設計思想（アーキテクチャ）に基づいて独立性の

高い単位（モジュール）に分解し、モジュール間を社会的に共有されたオープンなインターフェースでつなぐことにより汎用性を持たせ、多様な主体が発信する情報を結合させて価値の増大を図ること」と定義している。裏返してみると、オープン・アーキテクチャの形態をとる各 OS は、それまで個別のフォントフォーマットを採用していたが、オープンモジュールとなったフォントフォーマットを採用することで価値を増大させていく戦略をとったことになる。

図 4 は、OS 上の 1 モジュールであるフォントと利用者との関係を簡略表現した図である。パーソナルコンピュータにおける文字表示は、利用者が入力した情報を Input Method により文字コード化し、それを Application が受け取って、Output Method にレイアウト情報を渡し、Output Method は文字部分についてはフォントからの情報を基にラスタライズして、モニタやプリンタなどの出力装置に送ることによって、利用者に可視情報としての文字を提供している。

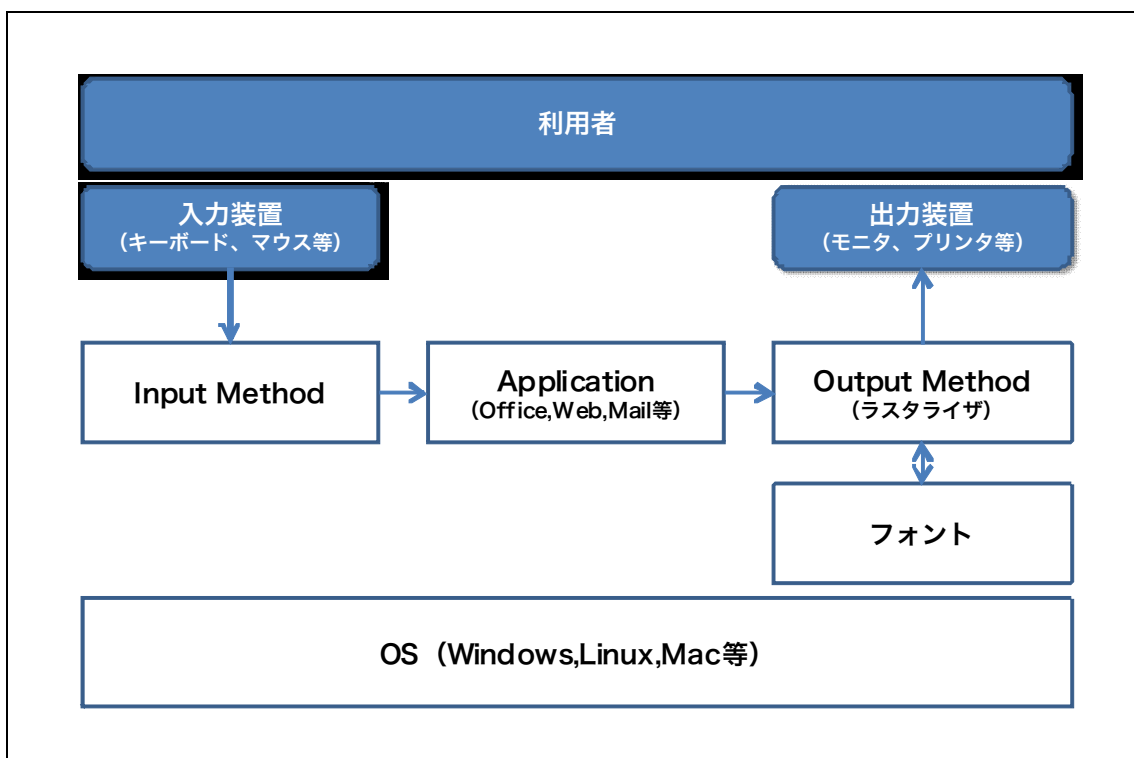


図 4 モジュールとしてのフォント

2. 6. 3 日本語フォントの品質

日本語フォントの製作には、書体デザインの知識や経験に加え、1 書体の持つ文

字数が膨大であることから多大な労力を必要としてきた（林、1996）。

日本語フォントのデザイン条件としては、文字の安定感、親しみやすさ、文字のつながりと流れ、錯覚（錯視）の利用、字画による画線の太さと間隔、黒さの均一性、印刷適正、などがある。特に日本語の場合は、横組だけでなく縦組をすることから、組版バランスを考慮し、記号類などは縦書き用の字形を用意することがある。読みやすさを考慮するとひらがなと漢字は一目で区別できることが必要（佐藤、1963:134）、といったノウハウが必要とされる。

また、漢字の字形デザインを行う上で、漢字の成り立ちについての知識も要求される。さらに、OpenType フォントフォーマットは、レイアウトに関する情報を Feature テーブルとしてフォントに保持することができるため、OS やアプリケーションとの調整が必要となる。

高品質フォントとは、上記の質的要素を充分考慮して作成され、標準に準拠した文字集合を搭載したフォントである。市販の日本語高品質フォントは、1文字当たりの制作単価が1万円前後といわれており（小宮山、2003:13）、標準化されている文字集合のフォントを制作するためには莫大なコストがかかることがわかる。

現在、日本語フォントとして利用することのできるフォント数は、有償のフォントからフリーフォントまで合わせると数千～数万フォントとも言われる¹¹。高品質なフォントは基本的には販売目的で作成されており、高価である。フォントをデザインの一部として商用目的で使用するユーザーはフォントを購入するという概念を持つが、一般のPCユーザーにとって、フォントはあつて当たり前のものであり、自分がどのようなフォントを所持し、どのような権利を保持しているかには無頓着である。OS にバンドルされている日本語書体を使用しているか、あるいは購入したアプリケーションにバンドルされているフォントを使用しているかのどちらかであろう。いずれの場合もフォントは無償ではない。ユーザーはOS やアプリケーションソフトウェアに対価を支払っており、その価格にフォント料金がアドオンされている。そこにはフォントに対する使用許諾が存在している。

¹¹ 小宮山(2008b:10)は、2008年現在に市販されているOpenTypeフォント1,650書体の書体分類をおこなっている。

2. 7 文字コード

日本語の文字をコンピュータ上で扱うためには文字コード体系というものが必須である。コンピュータによる日本語処理の黎明期はメーカー独自に搭載する文字を決め文字コードを割り振り、それに合わせてフォントを制作していた。しかし、パソコン通信による他機種間での情報交換が可能となり、さらに、専用ワードプロセッサからパソコン用のワープロソフトの普及へと進む中で、情報交換性を確保するために、日本語の文字コードが必要となった。そして、日本語のために 1978 年に日本工業規格 (JIS) によって制定された符号化文字集合が JIS X 0208 (旧 JIS C 6226) である。

日本語は表記に、表意文字である漢字、表音文字であるひらがな、カタカナ、さらにはラテン文字や記号等を使用する世界でも稀な言語であり、コンピュータ上で扱う文字も膨大な数となる。

2. 7. 1 JIS X 0208

日本語のために 1978 年に制定された符号化文字集合が JIS X 0208 である。JIS X 0208 は、6879 文字を含む主として情報交換用の 2 バイト符号化文字集合を規定する日本工業規格であり、各文字に区点という番号が割り振られている。JIS X 0208 は、1978 年に 6802 字の文字集合で制定され当初は JIS C 6226 とされたが 1987 年に JIS X 0208 という名前に変更された経緯がある。本論文では、1978 年に制定された JIS C 6226 を統一して JIS X 0208 として表記する。

JIS X 0208 が 1978 年に制定されて以降、1983 年の改正で 75 字追加と 300 字の字形変更、1990 年の改正で 2 字追加と 225 字の字形変更、1997 年の改正では追加や字形変更は行われず規格票の整備が行われた (表 4)。一連の改正における文字の追加は、内閣から告示される人名用漢字別表¹²の改正に関係している (安岡・安岡、1999:30)。

¹² 戸籍法 50 条・戸籍法施行規則 60 条によれば、子の名に用いることのできる漢字 (人名用漢字) は、「常用平易」でなければならないとされ、常用漢字表に掲げる漢字と人名用漢字別表に掲げる漢字に制限されている。2009 年 12 月現在の人名用漢字別表：<http://www.moj.go.jp/MINJI/minji86.html>

表 4 JIS X 0208 規格の推移

規格名	文字数	推移内容
JIS X 0208 : 1978	6,802 字	JIS X 0208 制定
JIS X 0208 : 1983	6,877 字	75 字追加と 300 字の字形変更
JIS X 0208 : 1990	6,879 字	2 字追加と 225 字の字形変更
JIS X 0208 : 1997	6,879 字	規格票の整備

JIS X 0208 を文字コードとして用いるための実用的方法は、日本語 EUC、ISO-2022-JP、Shift_JIS の 3 種の符号化方式である。

ここで文字コードについて補足すると、小形(2003)による“文字コード=文字セット(文字集合)+符号化方法(エンコーディング)”がわかりやすい簡略化表現といえよう。文字セットとはどういった文字がそこに収録されているかであり、符号化方法とはどういう方法、順番で文字に符号を対応つけるか、ということになる。しかし実際に“文字コード”という場合には、これらを混同して使用している側面は否めない。

パソコン上で所望の文字が表示できていない“文字化け”という現象の原因の 1 つは、同じ JIS X 0208 でも改正により同一の区点番号に異なった字形が割り振られてしまったことにあり、もう 1 つは 3 種の符号化方式の誤認識によって起こるものである。

2. 7. 2 JIS X 0213

次に、JIS X 0208 では文字が足りないという社会的要請をうけて 2000 年に 11,223 字を規定した JIS X 0213 が情報交換用の 2 バイト符号化文字集合を規定する日本工業規格として制定された。

日本語での日常的なドキュメントであれば JIS X 0208 の文字レパートリーだけで充分とされるが、人名、地名、諸外国語を取り扱うには文字が足りないという議論となった。文字の不足を補うために JIS X 0208 の空き領域を PC メーカー各社は外字領域として活用し混乱を招いた経緯がある(小形、2003:285)。メーカーやユーザーが空き領域を独自に使用した文字は外字と呼ばれ、特に JIS 符号化におけるメーカー外字は機種依存文字とされる。

表 5 に JIS X 0213 規格の推移を示す。JIS X 0213 は、2000 年に JIS X 0208:1997 を拡張する新しい文字コードとして、JIS X 0208:1997 の 6,879 字に 4,344 字を追加し合計で 11,223 字を規定している。これを JIS X 0213:2000 という。表記方法としては、JIS X 0208 まで用いられていた区点に面を加え面区点となり「面-区-点」で行う（日本規格協会、2000）。

さらに 2004 年には、JIS X 0213:2004 として 168 字の例示字形の変更と 10 字の追加を行った（経済産業省、2004）。このうち 168 字の例示字形の変更は、国語審議会の答申「表外漢字字体表¹³」の印刷標準字体に示されていた字形と JIS X 0213:2000 の例示字形が異なっており、これを合わせるための変更である。10 字の追加は、後述する国際標準規格 ISO/IEC10646 と Unicode において、JIS X 0213:2000 で符号化されている文字と包摂関係にある異体字に別の符号が与えられており、国際標準規格に合わせるために異体字を追加したものである。

表 5 JIS X 0213 規格の推移

規格名	文字数	推移内容
JIS X 0213:2000	11,223 字	JIS X 0208:1997 に 4,344 字を追加した新しい規格として JIS X 0213 制定
JIS X 0213:2004	11,233 字	168 字の例示字形の変更と 10 字の追加

ここで、標準化において混乱を招きがちな用語の定義を日本規格協会（1997）の JIS X 0208 規格票をもとに整理する。

字体：図形文字の図形表現としての形状についての抽象的概念。

書体：字体の具体化に際し、視覚的な特徴となって現れる一定のスタイルの体系が書体。篆書・隸書・楷書・行書・草書に加え、印刷文字における明朝体、ゴシック体、正楷書体、教科書体等が該当する。タイプフェイス。

¹³ 表外漢字字体表は、一部の印刷文字字体に見られる字体上の問題を解決するために、常用漢字表の制定時に見送られた「法令、公用文書、新聞、雑誌、放送等、一般の社会生活において表外漢字を使用する場合の字体選択のよりどころ」を示そうとするものである（国語審議会、2000）。

字形：字体を手書き、印字、画面表示などによって実際に図形として表現したものの。一つの字体には無数の具体的かつ可視的な字形が存在する。一つの字体についての字形の異なりはデザインの差である。

包摂 (unification)：ある文字の複数の字体を区別せずに、それらに同一の符号位置を与えること。包摂の規則を包摂規準という。JIS X 0208 の包摂基準には、以下の 6 項目の規準と具体的な字形 (図 5) が示されている。

- a) 方向・曲直などの点画の性質による違い
- b) 2 点画の接触交差関係の違い
- c) 2 点画の結合分離の違い
- d) 1 点画の増減の違い
- e) 類型の統合
- f) 筆法の簡化の違い

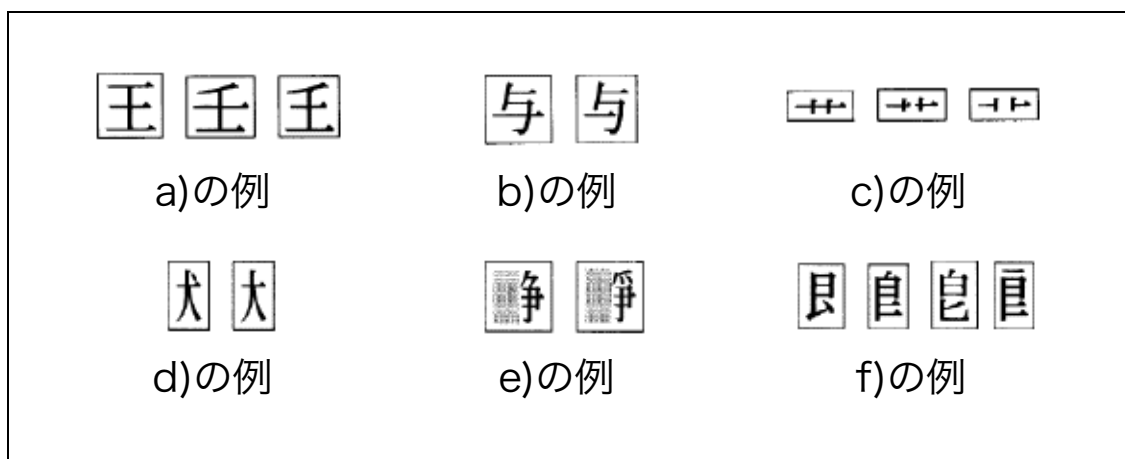


図 5 JIS X 0208 の包摂基準の例

例示字形：規格票を作るにあたって具体的な字形によって印刷する必要があり、その際に使用される字形。本来は、包摂基準により別の字形に置き換えられることを前提としている。

JIS X 0213：2004 での字形変更に対しては、文字コード規格は、文字の符号化表現を定めるものであり、例示字形は何ら規範的な役割を持たない、ことを謳っていながら例示字形の変更を行ったため、批判的な意見が寄せられた (小形、2004)。

また、JIS X 0213 には包摂関係にある別の字形にコードが割り振られている漢字

が 60 余個採録されており、規格として一貫性に欠く部分が存在するのも確かである。

2. 7. 3 ISO/IEC10646 と Unicode

ネットワークを基本とした情報化社会では、符号化文字集合が定まっていないと、論文等のドキュメント、メール、Web などでのコンピュータやネットワークを使った情報交換が正確に行えなくなる。符号化文字集合がばらばらに複数存在するとコミュニケーションミスやコミュニケーションコストの増大の要因になるといえる。

また、インターネットの急激な拡大により、国際的に通用する符号化文字集合が必要となった。世界中の言語に用いるあらゆる文字を統一的に扱おうとするのが国際標準規格 ISO/IEC10646 Universal Multiple-Octet Coded Character Set¹⁴（以下 UCS）である。

一方、Unicode は、すべての文字に固有の番号を付与し、プラットフォーム、プログラム、言語に依存しないことを目的として Unicode Consortium により作られた文字コード標準である（The Unicode Consortium、2006）。Apple 社、Microsoft 社、Hewlett-Packard 社、IBM、日本からは株式会社ジャストシステムなどが会員として参加している Unicode Consortium は、最新のソフトウェア製品と標準においてテキストを表現することを意味する“ユニコード標準”の構築、発展、普及、利用促進を目的として設立された非営利組織である。つまり、実際に自社製品に実装をする立場からの標準化活動であり、多くの会員企業はもちろん、産業界において Unicode 標準が採用されている。符号化方式としては、UTF-8、UTF-16、UTF-16BE、UTF-16LE、UTF-32、UTF-32BE、UTF-32LE の 7 種が定義されている。

UCS と Unicode とは「文字コードとしては同じ」とされている。UCS と Unicode は、両者が乖離しないことが合意事項になっており、現実には UCS に採録が決定した文字を Unicode も採録している。ただし、佐藤（2000）によると、UCS の規格票が単に文字コードであるのに対して、Unicode 規格として販売されている Unicode Book¹⁵には、いろいろな補助情報や実装上のことなども記述されており、

¹⁴ ISO/IEC 10646 - UCS : <http://std.dkuug.dk/jtc1/sc2/wg2/>

¹⁵ Unicode 規格については、最新版が Web に掲載されているが、印刷物としても発行されておりこれを「Unicode Book」と称している。2009 年 12 月現在、Web

UCS は UnicodeBook が言う実装をしなくても適合条件さえ満足すれば十分であるから、その点は UCS と Unicode で異なる可能性が残っているが、それも現在は顕在化していない。Unicode においても漢字の包摂規準 (Unification Rules) が定義されている (The Unicode Consortium、2006)。

また、日本工業規格である JIS X 0221 は、UCS の国際一致規格である。

2. 8 日本語環境の問題

日本語環境においては、符号化文字集合として扱う文字数が多い上に国内・国際の文字規準に変更が加わっていくため (図 6)、1つのフォントは一度作成すれば永続的に使用できるわけではなくメンテナンスが必要となる。今までにも規格内の変更や規格間の不整合が存在しており、文字化けのような情報交換上の混乱を生じさせている。もちろん、各 OS、アプリケーションが扱う符号化文字集合が変わると言うことになれば、InputMethod が変換に使用する辞書の拡張も必要となるし、OS やアプリケーションレベルでの実装にも影響を及ぼす。

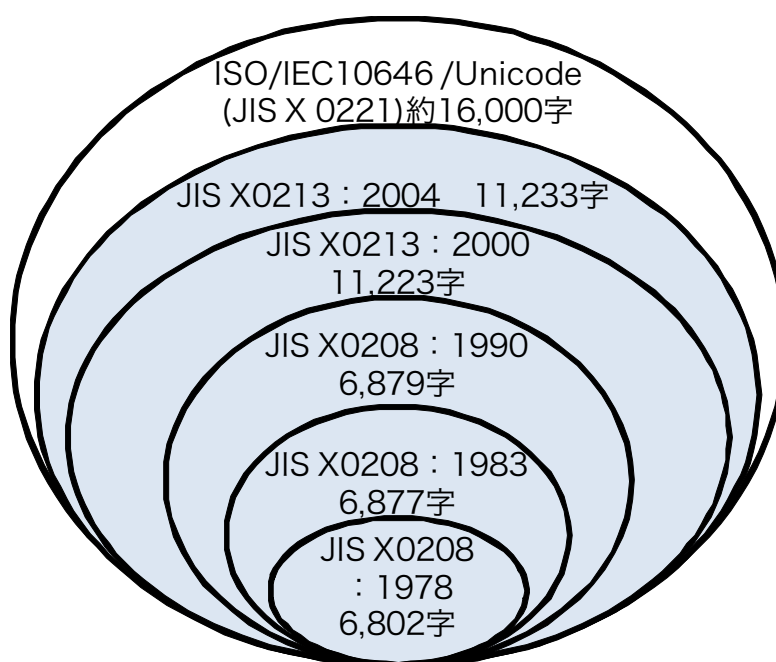


図 6 日本語符号化文字集合の変遷

上の最新版は Unicode 5.2.0 (<http://www.unicode.org/versions/Unicode5.2.0/>を参照)、Unicode Book は 2006 年に発行された「The Unicode Standard, Version 5.0」が最新版となっている。

IT産業の歴史の中で、情報通信基盤となる日本語フォントは有償でサービスを提供するOSベンダやSIベンダが独自に調達することで、各OSやSIベンダの提供するクローズドな日本語環境での整合性をとってきた。しかし、インターネットが普及し、オープンソースソフトウェア（以下OSS）における開発が推進される中、フォントについてもオープンであることが求められるようになってきた。

これまで、日本語環境の基盤整備において文字コードについては、国内外で標準化の活動が行われてきたが、フォントについては、民間フォントメーカーによって維持管理されてきた。そして、図4のように、フォントがオープンモジュールとなることで、フォントとラスライザ間の調整に負荷がかかることなく、文字表示を実現している。しかし、実際のところは、OSやアプリケーションの違いにより文字表示の品質に差異があり、レイアウト上、文字が欠けたり重なったりするなどの問題が出ている。これは、フォントとモジュール間のコミュニケーションが円滑に行われていないことが原因であり、真にオープンではないための情報不足、あるいは、オープンモジュールの持つ冗長性などが原因として考えられる。

一方、日本政府がOSSの普及を推進する中、OSSにおける開発環境、異なるOS間におけるマルチOS環境で自由に利用できる日本語フォントがないという問題が明らかとなった。OSS活動さらには日本のソフトウェア産業の活性化において、オープンな日本語フォントの不在が大きな障壁となっていた。

第三章、第四章では、このオープンな日本語フォントの不在という問題を解決するため独立行政法人 情報処理推進機構（以下IPA）が取り組んでいる「IPAフォント」に着目し、筆者がメンバーとなって推進した「IPAフォント」を日本語パブリックフォントと位置づけ、そのオープン化の提案と意義、さらには日本語パブリックフォントのライセンスのあり方について述べる。

さらに、文字については、社会的、文化的側面でいつの時代にも課題が存在している。その1つとして、符号化文字集合が拡大しても継続する外字問題がある。北川（2007:33）が日本の地方自治体での技術的課題に外字処理をあげているように、国や地方自治体で扱う人名・地名などの中に外字問題が存在する。外字問題は2通りあり、1つは包摂規準の範囲で符号化文字集合に収録されているが、字形を区別できない文字（＝異体字）の処理問題であり、もう1つはそもそも符号化文字集合上に符号位置を持たない文字（＝外字）の処理問題である。今後、電子政府や電子

自治体を推進していく上でテキストデータの互換性、相互運用性、検索性を高めるために外字問題への取り組みは大きな課題といえる。

第五章では、この外字、異体字問題への取り組みの1つとしてグリフデータベースについて述べる。

第三章 日本語パブリックフォントに関する検討

3.1 はじめに

ICTにおいて文字を表示するための「フォント」は欠かせない存在である。ソフトウェアにおける日本語環境の整備を論じる際、日本語フォントが当然あるものとして展開されるが、実際には多大なコストをかけて開発された日本語フォントが無ければ日本語環境は成立しない。

コンピュータ上で扱う日本語フォントには、1万字以上という字形集合の大きさ、文字コード、異体字・外字処理、知的財産権、可読性などのクオリティ維持のためには多大な開発コストが必要などといった多くの課題を抱えている。

インターネットの普及により国際化が叫ばれる中、日本のIT産業が競争力をもつて開発を推進するため日本政府はOSS普及推進を進めてきた。しかし、OSSにおける開発環境、異なるOS間におけるマルチOS環境で、日本語の処理、表示を行うためには日本語フォントは必須であるが、これまで、OSSで利用可能な日本語フォントは存在していなかった。オープンな日本語フォントの不在がOSS活動さらには日本のソフトウェア産業の活性化の大きな障壁となっていたことから、独立行政法人 情報処理推進機構（以下IPA）は、2003年より「IPAフォント」をある制限のもとで提供してきた。IPAフォントは、商用フォントと同等レベルの字形集合を保持し印刷にも対応できる可読性を追求した高品質フォントである。しかしフォントを情報通信基盤として明確に位置づけ、それを基盤として整備するためにどのような条件が必要か、どのような整備の方針がありうるかを検討した研究はなく、IPAフォントは、十分な管理体制およびライセンス体制にはなかった。本研究はこれを試みるものである。

本章では、技術と社会環境の変化により公共フォントとして日本語パブリックフォントという新たなドメインが必要とされてきた状況を整理し、IPAフォントでのパブリックフォントの整備事例を提示し考察することで、今後の日本語パブリックフォントの課題を提示する。

本研究は、筆者が2007年から現在までIPAの研究員として在籍し、IPAフォン

ト (Ver.002) の公開 (2007 年 10 月 1 日に公開実施) の担当者として活動する中、公開方針の検討にあたって調査・研究を行った結果を基に、筆者が日本語パブリックフォントの課題を検討したものである。この研究は IPA の了解を得て、筆者を第一著者として学術雑誌に投稿・掲載された。

この研究で得た成果は、初めての日本語パブリックフォントと呼べる IPA フォントの今後の運用に役立てることができる。さらに、広く本研究成果を公開することで、未整理であった日本語フォントのポジショニングを明確にし、今後の OSS をはじめとするソフトウェア産業の育成にあたっての基盤整備に必要な課題を提示し、さらなる議論が展開されることが期待できる。

3. 2 フォントのモジュール化・オープン化

フォントを情報通信基盤として整備するうえでの重要な着眼点は、モジュール化・オープン化である。

Carliss Y. Baldwin, Kim B. Clark (2000=2004) はパーソナルコンピュータにおいてモジュール化の発想がいかに重要であり、それがいかにイノベーションを創出し、ソフトウェア産業、シリコンバレー、ネットワーク経済を生んだかを示した。国領 (2003:72) は多数の技術の複合したシステムにおける自律・分散・協調の基盤を支える重要な設計思想をオープン (開かれた構造) 化とした。オープン化の前提としてモジュール構造がある。

しかし、モジュール化はイノベーションを加速させる側面を持つが、モジュールは共通のインターフェースを持つため冗長性を内包しており (青島・武石, 2001:43)、システムが最適なパフォーマンスを得るためには構成要素間の情報を仲介する機能が重要となる。

日本政府が OSS を推進する理由には、ソフトウェア技術の中での知識共有によるイノベーションの促進があげられる (田代, 2006:540)。一国の持続的な成長、発展を実現するために、イノベーションの促進は、政府の政策にとって重要なテーマである (後藤, 2001:9)。社会構造の変化の中で、日本経済の成長・発展を促すためのイノベーションの促進の中で求められているオープンな日本語フォントを定義することは、情報通信基盤の整備と言える。プロセス全体を解決するその問題点を明らかにすることが、イノベーションへの道の具体的な活動 (宮原, 2005:243) で

あり、日本のソフトウェア技術の中での日本語問題を明らかにすることは、イノベーション促進活動とおける。

本論文における「基盤」とは、インフラストラクチャー (infrastructure) を指しており、社会資本と同義語として用いられることが多い。社会資本は通常政府によって整備される (羽田、2010)。また、社会資本は社会的間接資本または社会的共通資本とよばれることもあるが、これは、社会資本が生産活動や消費活動などの経済活動一般の基礎となり、財・サービスの生産に間接的に貢献することを意味する (大塚、2010)。本論文で扱うオープンな日本語フォントは、OSS 推進により日本経済の成長・発展を促すという経済活動の基礎として、政府から提供された公共財として捉えることができる。

「2. 6 日本語フォント」でみたように、日本語フォントは、パーソナルコンピュータの出現とネットワーク社会化により、データでの情報交換を可能とする環境への対応として、標準化・規格化の必要に迫られ、結果としてクローズドなモジュールからオープンなモジュールへと変化していった。

イノベーション加速を前提とした公共フォントのあり方を検討するにあたってはフォントというプラットフォームのデザインを考えるべきであろう。プラットフォームとは「第三者間の相互作用を促す基盤を提供するような財やサービス」であり、プラットフォームに適度な制約とルールが存在することで、創発的な価値創造を促す (国領、2003:114-117) とされる。フォントは、プラットフォームとしては未整備であるために、創発的な価値創造にとってボトルネックとなってきたように見える。従って、これからの公共フォントを考える上で、適度な制約とルールを見いだすことが重要である。日本語フォントが最適に機能するための情報を仲介するシステムをデザインすることが、日本のソフトウェア産業のイノベーション促進へとつなげるための課題である。

3. 3 日本の OSS 政策とフォント

3. 3. 1 OSS の社会的役割

OSS は、1998 年に設立されたオープンソースイニシアチブ¹⁶ (以下 OSI) が推進

¹⁶ Opensource.org、Open Source Initiative : <http://opensource.org/>

する新たなソフトウェアの開発スタイルである。Raymond (1999) は、旧来の組織的な開発スタイルを「伽藍 (Cathedral) モデル」とするのに対し、誰もが自由に集まり、好き勝手に、他方で協調しながら開発を進めるモデルを「バザール (Bazaar) モデル」と名付けた。OSS はソースコードを開示しつつ「バザールモデル」で開発を進めることで、様々な開発者の知見を効果的に一つのソフトウェアに集めることができる。世界レベルでのオープンな開発コミュニティの中で、メーリングリストや Web で情報公開をおこなうことで、貢献や贈与という開発者自らの開発モチベーションを維持し効率的かつ継続的なソフトウェア開発を可能にしていると捉えることができる。

また、OSS ライセンスは、誰もが一定の条件に従えばソースコードを自由に利用 (複製・配布・改変) できることが重要なポイントである。OSS であるかどうかは、OSI による定義である The Open Source Definition (以下 OSD) を満たしているかどうかであり、ソフトウェアの著作権者は、ライセンスによってそのソフトウェアをどう扱ってほしいかの意志表明をすることになる。OSD では「フリーであるかどうか」をベースにした OSS の条件のガイドラインが設けられており、OSS 利用者は利用の際に著作権者とライセンスについて調整することなく安心して利用できる。つまり、OSS とは、図 7 に示すように、OSI が OSD を満たしていることを確認した上で承認した OSS ライセンスのもとで公開されるソフトウェアである。

OSS で最も有名な例に、1991 年に Linus Benedict Torvalds が開発した Linux OS がある。その他にも Apache、Tomcat、Emacs、PostgreSQL、MySQL、JBoss といった OSS ミドルウェアや業務アプリケーション、システムの開発環境においても OSS の活用が進んでいる。これらの OSS は、ソースコードを公開するとともに広く多くの意見を取り入れることから、オープンスタンダードを採用していることが多い。つまり、OSS の採用はオープンスタンダードと直結することで、技術的なリスクを回避することにもなる。

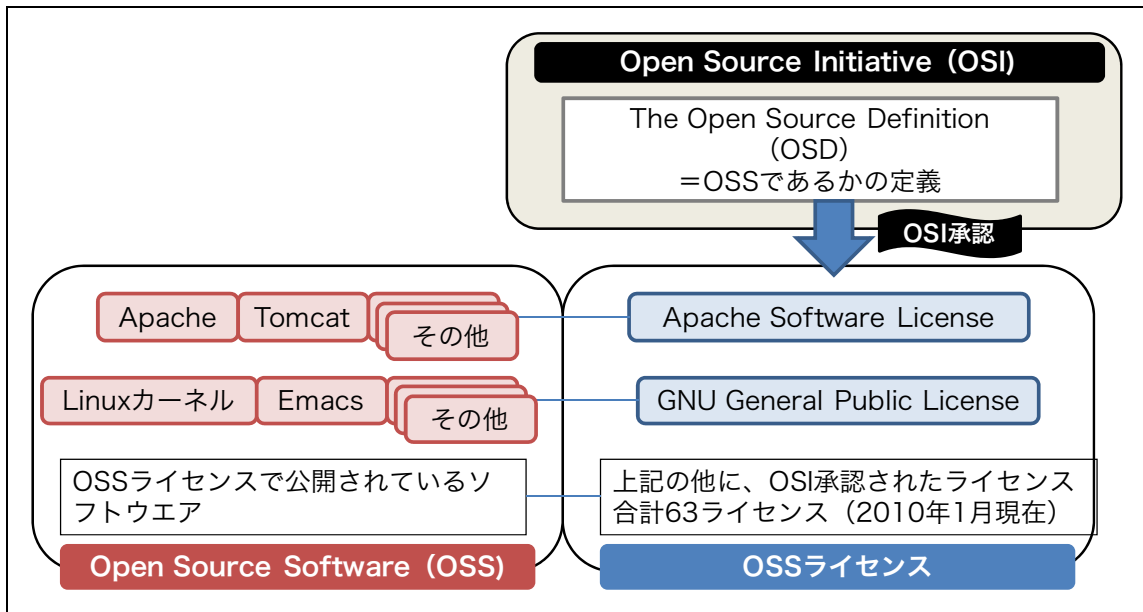


図 7 OSS と OSS ライセンスの関係

OSS の普及とともに、世界各国で公的組織が OSS を推進している。

欧州では 2000 年に開催された欧州委員会 EU リスボン会議で、2010 年間までに EU 全体をより豊かにしていく政策目標としてリスボン戦略を掲げ、その一環である ICT 戦略の中で PS-OSS (Public Sector & Open Source) プロジェクトを立ち上げた(北川、2007:18)。PS-OSS では公的組織を対象とした FLOSS (Free/Libre/Open Source Software) に関するプロモーションを進めている。

アジア各国でも OSS の教育への活用や政府主導で導入を推進する取り組みが行われており、特にソフトウェア産業でのベンダロックインによる貿易不均衡解消への対策としても OSS 導入に積極的である。

3. 3. 2 日本の政策の中の OSS 推進

2001 年、内閣に高度情報通信ネットワーク社会推進戦略本部 (IT 戦略本部) が設置され、同年には e-Japan 戦略が発表された。日本の情報政策史区分(砂田、2007)では、2001 年から現在までは「情報政策における省庁間連携モデル構築期」にあたり、情報政策はそれまでの通商産業省を中心に立案・実施された時代から内閣府を中心として各省庁間で連携する時代へと移行した。1960 年代から 1970 年代の情報政策は、国産コンピュータ産業政策として海外から「通産省モデル」と呼ばれて注

目されるほどの成功を取めたが、1990年代にはインターネットの普及などに伴い通産省モデルは機能しなくなった。

現在の高度情報通信ネットワーク社会において、情報政策もクローズドからオープンへの構造変革が求められたといえよう。

e-Japan 戦略は、すべての国民が IT の恩恵を享受できる知識創発型社会の実現を掲げ、超高速インターネット網の整備、インターネット常時接続の早期実現、電子商取引ルールの整備、電子政府の実現、新時代に向けた人材育成等の IT 基盤整備により我が国が 5 年以内に世界最先端の IT 国家となることを目標としてスタートした。

この e-Japan 戦略の中で OSS が取り上げられたのは 2003 年の e-Japan 重点計画 2003 からである。e-Japan 重点計画 2003 では、重点政策 5 分野がリストアップされ、その 1 分野である「電子商取引等の促進」において「OSS 市場の拡大」があげられた。OSS の開発・利用を促し、OSS に関係するビジネスを活性化させることで、ソフトウェア産業界全体の国際競争力の向上を図ることをねらいとしている。

さらに、2005 年の IT 政策パッケージ-2005 では、OSS のより迅速な活用促進をめざし、電子政府における OSS の活用促進、初等中等教育への OSS の導入、大学等での OSS を利用した高度 IT 人材の早期育成への取り組みを目標に掲げた。

これらの e-Japan 戦略の後押しにより、2003 年度から IPA を中心とした OSS 推進のための技術開発、導入実験などのプロジェクトが進められた。2006 年には、OSS 普及に向けた障壁となる事柄への解決に向けた業務を集中して行う組織として IPA 内にオープンソースソフトウェア・センター（現：オープンソフトウェア・センター）が設立された。

一方、2004 年 2 月、政府の声かけにより、企業、政府、学識経験者などで構成する OSS 推進普及団体「日本 OSS 推進フォーラム」が設立され、日本の主要 IT システムベンダが参加している。OSS の発展のカギ（桑原、2006:419）として、ユーザーの目覚め、ユーザーの潜在要求を捉えた OSS 側の先行性の確保、OSS コミュニティの自由かつ活発な活動をあげており、これらを日本 OSS 推進フォーラムでの活動推進目標としている。

3. 3. 3 OSS の産業育成に果たす役割

日本政府が OSS を推進する背景には、日本はものづくり国家としてハードウェアを中心とした産業において一定の評価を得ているが、ソフトウェアは産業として未熟な状態にあるため、その育成が課題となっていることが挙げられる。

日本政府が OSS に注目する理由には産業活性化がある（田代、2006:540）。ハードウェアの技術発展における過去の技術基盤の共有化は、特許などの知財の公開情報の共有によりうまく進んできたが、ソフトウェアは持続期間の長い著作権保護に守られており、知識共有によるイノベーションが起こりにくい傾向にあり、OSS の導入がこの問題に対する 1 つの方策とされている。

村上（2004:23-25）は、現在は特定 OS への過度な依存状態下にあるが、特定 OS に依拠しないための選択の自由の確保や、アプリケーションソフトウェア及びそこで処理されるデータ資産が特定の OS に依存するリスクの回避、をあげており、そのためにも政策上重要な目的として、単に特定 OS 以外の選択肢を増やすだけでなく、中身においてより優れた OS を作ること、OS 以外の分野においてもこうした問題を事前に解決するようなソフトウェアが作られることを、政府なりに支援することをあげている。

OSS を相互運用性の高い OS やソフトウェア開発の拠り所とし、国レベルで OSS 環境の促進をサポートしていくことは、日本発のソフトウェアが非常に少ないという現状をふまえても、情報化社会における日本の重要な産業政策といえる。

3. 3. 4 OSS 推進におけるフォントの役割

ソフトウェア上で文字を表示するためにフォントが不可欠であることは自明であるが、これまで情報通信基盤として日本の国家レベルで知的財産権を保有し国民が自由に利用できる日本語フォントは存在しなかった。

過去に日本語環境の整備や日本独自 OS の開発を目指したプロジェクトの中で平成フォントおよび GT 明朝といったフォントに着手したケースはあるが、これらは一定の制約がかかったフォントである。

平成フォント¹⁷は旧通産省工業技術院の電気・情報規格課のイニシアチブにより

¹⁷ 平成フォントについては澤田善彦『フォント千夜一夜物語』の平成フォントに

1988年に設立された「文字フォント開発・普及センター」が開発した日本語フォントである。当時、日本語フォント環境は整備されておらず、コンピュータ関連の機器メーカーやソフトウェアメーカーなどは、フォントが欲しくても入手できない状況にあった。大手コンピュータ・メーカー、大手電子機器メーカー、大手印刷企業、写植機メーカーなどが開発会員として出資して、平成フォントを共同開発し、各社のビジネスで使用可能とした。しかし、この平成フォントは開発会員の所有物であり、また現在「文字フォント開発・普及センター」は解散し、平成フォントの開発は終了している。

K. Sakamura (1987) に始まる TRON プロジェクトでは、世界のあらゆる文字を収録するというコンセプトの下、150万文字の収納可能とする TRON コードを設け、現在約 18万文字を収納した GT 明朝フォントを公開している。しかし、GT 明朝を使用するためには、パーソナルコンピュータ用に開発された BTRON 仕様 OS 「超漢字」が必須であるため、マルチ OS 環境で自由に利用できる日本語フォントとは成り得ない。

国家レベルでのフォントの必要性、つまり公共財としてのフォントの必要性についてはこれまでほとんど議論されていない。その理由を、日本語文字処理における IT パラダイムの転換から解釈することができる。メインフレームが主流だった時代には、提供ベンダにとってソフトウェアや関連サービスは、ハードウェアの付随品として取り扱われていた（伊藤、2007:10）。メインフレームが 1 社で整合のとれた製品を提供する時代には、日本語フォントも外部との互換を考慮せずクローズドな環境で表示・印刷できるものを調達すれば良く、国レベルでフォントを整備するニーズはきわめて低かったと言えよう。

しかし、インターネット時代に突入し、特に日本の産業政策として OSS を推進する中で、相互運用性の高い OS やソフトウェアとともに日本語文字処理に必須である日本語フォントが必要になった。これまでは、ほとんどの PC ユーザーは有償 OS ベンダや SI ベンダが独自に調達したフォントを日本語環境の一部として利用していたが、OSS では特定のライセンスに縛られないパブリックフォントという新たなドメインの日本語フォントが求められてきた。つまり、OSS 推進の中でオープンな

についての記述を参考にした：

http://www.jagat.or.jp/story_memo_view.asp?storyID=1476

日本語フォントの不在がボトルネックとなったと捉えることができる。社会資本は、民間企業にゆだねていたのでは、その供給がなされなかったり、不足したりするゆえに政府の手にゆだねられるのであり（大塚、2010）、日本語フォントについても、公共財として政府から提供されることが求められていた。

IPA フォントはこうした課題を解決するために構想された。

3. 4 海外における国家レベルでの OSS とフォント政策

漢字圏である日本、中国、台湾、韓国では、同一文字コードの漢字であっても字形が異なる文字が多いが、日本以外はビジネスシーンにおいて字形へのこだわりが薄いと言われている。

以下に、中国、台湾、韓国およびその他における国家レベルでの OSS 推進とフォント政策の状況について整理する。

3. 4. 1 中華人民共和国（中国）

1999 年、政府により公式に OSS 概念が持ち込まれ、2004 年には情報産業部中国 OSS 推進連盟が発足している（国際情報化協力センター、2008a）。中国語版 Linux 開発や、電子政府等の政府調達に国産 Linux を優先的に導入する動きがあったが、2008 年のサーバ分野での LinuxOS シェアは 20%あるもののデスクトップ分野は 2%と低く、WindowsOS のシェアが拡大傾向にある。Linux 系ソフトウェアは、政府購入計画を除いて民間のビジネスユーザーに採用されるケースは少ない。

国内で使用するフォントはすべて国家機関である全国情報技術標準化技術委員会（NITSTC）¹⁸による審査・登録制がとられている。また、NITSTC は国家標準となるビットマップフォントを保有しており、ライセンス契約に基づく企業への有償の提供を行っている。

漢字については、日本と同様に異体字問題は存在するが、情報交換性の無いクロードなフォントとして外字を使用することで問題ないとされており、かつ国家が

¹⁸ NITSTC については、以下を参考にした。

ダイナコムウェア株式会社「全国情報技術標準化技術委員会規約（和訳）」：

http://www.dynacw.co.jp/img/license/Articles/CITSTC_Artc.htm >

「中国新文字コード規格 GB18030」：

<http://www.dynacw.co.jp/license/gb18030/#Windows>

外字の使用を制限していることもあり、使用できる漢字が足りないという問題は重要視されていない。

このように中国ではフォントは国家レベルで管理されているが、フリーまたはオープンというドメインのものではない。

3. 4. 2 台湾

2002年6月、台湾行政院はソフトウェア産業強化のため OSS 推進の実行機関として経済部工業局内に Free Software Steering Committee を設立、2002～2007年の OSS プランを発表した（国際情報化協力センター、2005）。2002年以降、OSS に対するコミュニティの関心や国際動向の高まりに伴い、メディアや法の専門家が OSS 分野に興味を示し始めている。

フォントに関しては、特に国家機関での統制や開発などは行われていない。台湾で使用される漢字は繁体字が主とされるが、簡体字も中国とのデータ交換などで使用されることがある。漢字文字コードには台湾の独自コードである BIG5 が多く使用されており、BIG5 には 13,461 字が収納されている。台湾でも日本、中国と同様に異体字問題があるが、中国のように国家の規制もないことから、異体字の扱いについて問題視する声は高い。

国家政策とは別に、台湾のフォントメーカーである ARPHIC TECHNOLOGY 社が 1999 年に 4 種類の自社フォントをフリーフォントとして提供している。配布にあたり、独自のライセンスである Arphic Public License¹⁹を定義しており、フォントの改変・配布の自由が認められている。

OSS の中国、台湾、香港などの漢字圏コミュニティでは、Arphic フォントと Arphic フォントを改変したフリーフォントが使用されている²⁰。

ARPHIC TECHNOLOGY 社が高品質なフォントをどういった経緯でフリーというドメインに投入したかについての目的、経緯は明確ではないが、Arphic フォントの存在により繁体字、簡体字については使えるフォントがないという状況には陥つ

¹⁹ ARPHIC PUBLIC LICENSE :

<http://ftp.gnu.org/non-gnu/chinese-fonts-truetype/LICENSE>

²⁰ 「Unicode Font Guide For Free/Libre Open Source Operating Systems」を参考にした：<http://www.unifont.org/fontguide/>

ていない。

3. 4. 3 大韓民国（韓国）

2002年12月、政府はLinux開発を積極的に推進すると発表した。2003年にはOSS活性化政策を導入している（国際情報化協力センター、2008b）。

フォントに関しては、特に国家機関での統制や開発などは行われていない。

韓国における文字使用はハングルが基本であり、漢字は中学校、高等学校での漢字教育用と指定されている基礎漢字1800字が定義されている。文字数としては、Unicodeでは現代ハングル音節文字11,172文字が定義されている。

多文字でありながら、韓国語フォントは、個人の制作によるGPLライセンス（GNU General Public License）²¹のフリーフォントが配布されている。

3. 4. 4 その他

漢字圏以外で国家レベルでのフォントの制作や管理を行っている事例を挙げる。

タイでは、2003年6月、ICT省と科学技術省と民間セクタでのOSS開発・促進に合意し、政府R&D機関NECTECがタイ語対応のディストリビューションを開発しており、OSSの活動の中でフォントの開発も行っている（国際情報化協力センター、2008c）。

インドでは2007年に政府IT省が、インドの主要7言語のソフトウェアツールやフォントを含んだCDを一般にリリースした（国際情報化協力センター、2008d）。

タイ、インドともにアジア圏の中ではIT技術力が高い国であり、国策として情報通信の基盤となるフォントを提供していることがうかがえる。

3. 5 IPA フォント公開の経緯

3. 5. 1 IPA フォント（Ver.001）公開の経緯

OSSの中でもとくに活動の盛んなLinux OSは、多くのOSSの活動の基盤となるOSである。その中で、Linuxをデスクトップ使用の際の操作メニューやドキュメントの表示・印刷に日本語フォントを使用するニーズは非常に高い。しかし、

²¹ GPLライセンスについては、「GNU 一般公衆利用許諾契約書（GNU General Public License）」が公開されている：<http://www.gnu.org/licenses/licenses.ja.html>

日本語フォントの開発には膨大なコストがかかるため、権利関係が明確でかつ高品質なフリーフォントが存在しないという問題を抱えていた。Linux などの OSS 環境上で日本語アプリケーションを開発する際にも、同様に日本語フォント問題が出てくる。当然であるが、Windows や Mac OS X にバンドルされているフォントは、それぞれ購入した OS 上でしか使用できず、異なる OS へのコピーは使用許諾に反する違法行為となる。

フリーな日本語フォントとしては、2000 年 11 月に公開された東風（こち）フォントがあり、2003 年には日本語の Linux ディストリビューションのほぼすべてに収録されていた。しかし 2003 年 6 月 15 日に権利侵害の問題が取り上げられ、フォント制作活動の終了に至っている。東風フォントは、開発当時にフリーフォントと認識されていた 32 ドットのビットマップ・フォント（渡邊フォント）を元として修正を加えアウトライン化した TrueType フォントである。しかし、元となった渡邊フォントは、(株)日立製作所デザイン研究所（現在の日立製作所 デザイン本部）と(株)タイプバンクが 1980 年代に共同制作した商用製品「日立-TB 32 ドットフォント明朝体」を無断で複製したものであることが発覚した。本問題が日本の現行法で「著作権侵害」に該当するか否かの結論は出ていない。関係者の議論・交渉により、2003 年 9 月 29 日には「32 ドット明朝体の権利を使用したフォントの制作、公開、配布を“一定の条件”のもとに認める。」という決定ができたものの、条件には、契約締結や権利使用の明記、非営利使用に限定した配布、が提示されていた。東風フォント制作者は諸権利を尊重し、決定に同意するものの「自由なフォントが欲しい」という希望を満たせなくなったため開発動機を維持できない、ということで 2003 年 10 月 21 日に活動終了を宣言した²²。

²² 東風フォントに関する一連の経緯は以下を参考した。

・フリーのフォントに権利侵害の問題が見つかる 【6 月 27 日に追記】Linux ディストリビュータらは対応に苦慮：

<http://itpro.nikkeibp.co.jp/free/LIN/NEWS/20030624/1/?ST=oss>

・フォント無断複製問題の影響により、東風フォント制作活動終了：

<http://opentechpress.jp/news/03/10/22/022214.shtml>

・東風フォント製作中止に：<http://slashdot.jp/article.pl?sid=03/10/22/0811250>

・緊急作成 --- 32 ドットビットマップフォントの無断複製について：

<http://wiki.fdiary.net/font/?stolenbitmap>

・東風フォント制作活動終了のお知らせ：

http://www.on.cs.keio.ac.jp/~yasu/jp_fonts.html

これにより、OSS環境で自由に利用できる日本語フォントがなくなってしまったため、急遽日本語フォント調達の動きが活発となった。

IPA（2003年当時は情報処理振興事業協会、2004年1月5日より独立行政法人情報処理推進機構が業務を承継）においても、東風フォント問題はIPAが支援する「オープンソフトウェア活用基盤整備事業」の成果物となる日本語環境整備を中心としたソフトウェアの研究開発に多大な影響を与えると判断した。今後のソフトウェア研究開発と、日本語電子文書の相互互換性の確保のために、日本語フォントは重要である、という認識のもと、2003年11月28日に日本語フォントを競争入札²³という形で調達した。2003年12月18日、IPAが全権利を保有する高品質な日本語フォント5書体が誕生した。このフォントはIPAフォント（Ver.001）とされ、JIS X 0208:1997準拠のTrueTypeフォントである。

IPAフォント（Ver.001）は、IPAが支援した「オープンソフトウェア活用基盤整備事業」で開発されたソフトウェアに同梱する形でのみ配布を行った。実際にIPAフォント（Ver.001）の使用を許可されIPAフォントを同梱したソフトウェアは、GRASS、Knoppix、OpenPrintingの3件であった。2009年12月現在、3件のうちKnoppix²⁴、OpenPrinting²⁵の2件が公開されている。

3. 5. 2 IPAフォント（Ver.002）公開の経緯

IPAフォント（Ver.001）が公開されてのち、JIS X 0213：2004が規定された。IPAフォントを最新の規格準拠とするため、2006年11月30日にIPAフォント（Ver.001）から、JIS X 0213：2004に対応する差分の字体変更および文字追加を行う「IPAフォントの拡張」に関する調達をおこない、これをIPAフォント（Ver.002）とした。しかし、IPAフォント（Ver.002）については、Ver.001の際の限定的な使用許諾ではなく、フォントの自由な活用と研究開発の推進という社会全体での活用を考慮した環境作りを目指すため、運用方法を検討するフェーズに置かれた。

一方で、東風フォント開発の活動が停止したのち、OSSコミュニティで自由に利

²³ IPA 公募・入札一覧：<http://www.ipa.go.jp/about/kobo/tender-20031128/>

²⁴ Knoppix：<http://www.alpha.co.jp/biz/rdg/ac-knoppix/>

²⁵ OpenPrinting：http://avasys.jp/product/linux_driver/opp2003.php

用可能な代替フォント開発²⁶が行われている。2004年6月21日に公開された「さざなみフォント」である。

さざなみフォントは、東京大学和田研漢字分科会で研究開発された和田研フォントキット（研究活動期間は1990年4月から1992年3月まで）を使用して自動的に生成されたフォントと、既存の改変自由なビットマップフォントを組み合わせた日本語 TrueType フォントである。フォントの自動生成は漢字の偏や旁のような部品をアルゴリズムによって組み合わせることで行うものである（田中ら、1995）が、研究用途のツールであるため、自動生成された字形は決して美しくはなく、OSS環境に Windows や Mac 環境と同様の可読性に考慮した自由なフォントが提供されているとは言えない状況が続いていた。

このような中で、2007年10月1日、JIS X 0213:2004 に準拠した IPA フォント（Ver.002）をより多くのユーザーの活用を促進するため、「一般利用者向け IPA フォント使用許諾契約書」²⁷を定め、IPA の Web サイトより配布を開始し、公開後2週間ほどで3万ダウンロードを突破した。

3. 5. 3 IPA フォント（Ver.002）の使用許諾の定義

IPA フォント（Ver.002）をリリースするにあたり、限られた使用に限定するのではなく、広く多くのユーザーに使用を許諾するためのライセンスを設定することが課題であった。

IPA は、ライセンスを定義するにあたり以下の3点に留意した。

- (1) 一般利用者にとっての、デジタルコンテンツでのフォント利用に対して最大の自由度を付与する。
- (2) 個人利用において、改変、修正を自由にする。
- (3) 提供する IPA フォントの配布は自由に行えるが、個人が改変や修正した派生フォントの配布制限をおこなう。

²⁶ 「東風フォント」の代替フォントとして開発中の「さざなみフォント」が公開：
<http://www.forest.impress.co.jp/article/2004/06/21/sazanami.html>

²⁷ 一般利用者向け IPA フォント使用許諾契約書：
<http://ossipedia.ipa.go.jp/ipafont/>

検討の結果、IPA フォントのライセンスは SIL Open Font License (OFL) ²⁸をベースとするが、派生及び独自フォントの流通は不可とする方針をとった。自由性を最大限取り込み、かつ既存のフォントベンダライセンスが持つファイルエンベッドやサーバ使用の不許可といったデジタル利用特有の曖昧さを排除した。但し、OFL の"Reserved Font Name"に関する条項は削除し、派生フォントの流通は不可とした。日本語の真のパブリックフォントというドメインを築き、明日のフォント市場と文化を維持しつつ、IPA フォントの品質を保つためである。派生フォント流通のニーズへの対応は、第四章であらためて検討をおこなう。

3. 6. IPA フォント公開の効果

IPA フォント (Ver.002) リリース後に、一般ユーザーと Linux ディストリビューターから得た反響をまとめる。

3. 6. 1 一般ユーザーからの反響

IPA は、IPA フォント (Ver.002) を 2007 年 10 月 1 日にリリース後 2008 年 8 月末までに 1839 人からのアンケート結果を得ている。アンケート項目は、IPA で標準的に使われているフォームであり、以下の項目である。

(1) ご利用形態 (必須・1つ選択)

個人/企業/研究開発機関/教育機関/官公庁/その他 の 6 つの形態からの選択式

(2) ご利用場所 (必須・1つ選択)

都道府県からの選択式

(3) ご利用になる主な OS (任意・複数選択可)

Linux / UNIX / Windows / MacOS / その他 の 5 つの OS からの選択式

(4) 申込者氏名 (任意)

(5) 申込者メールアドレス (任意)

(6) 所属団体名・部署名 (任意)

(7) ご意見等 (任意)

²⁸) SIL Open Font License :

http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=OFL

アンケートは、IPA の Web サイトから IPA フォントのダウンロードを行う際、「一般利用者向け IPA フォント使用許諾契約書」への同意を行ったユーザーが、ユーザーの意志によりアンケートフォームの Web ページへのリンクをクリックすることで記述されたものである。

表 6 に IPA フォントダウンロードユーザーの利用形態（アンケート（1）の回答）と利用する OS（アンケート（2）の回答）を示す。利用形態としては個人ユーザーが多く、利用 OS は Windows(52.9%)が一番多く、次いで複数 OS(21.9%)、Linux（11.3%）となった。

表 6 IPA フォントの利用形態と主な利用 OS

利用形態（総回答数 1839 件）			利用している主な OS(総回答数 1839 件)		
	件	%		件	%
個人	1504	81.8	Linux	208	11.3
企業	151	8.2	UNIX	13	0.7
研究開発機関	19	1.0	MacOS	120	6.5
教育機関	66	3.6	Windows	973	52.9
官公庁	5	0.3	その他	18	1.0
その他	19	1.0	複数 OS	402	21.9
未回答	75	4.1	未回答	105	5.7

自由記述方式であがった 331 件の意見の内、199 件（60%）は IPA フォント提供に対する感謝とクオリティの高いフォントに対するプラス評価であった。OSS やマルチプラットフォーム環境で利用可能な可読性の高いフォントが今まで無かったため非常にありがたいという感想が多く、公的機関で質の高いフォントを提供・管理することを評価する意見も多数見られた。

一方、62 件（18%）の要望、苦言が寄せられており、内容を分類すると、1. フォントのバリエーション追加要望（20 件）、2. ライセンスに対する要望（13 件）、3. IPA フォントのデザインに対するコメント（12 件）、4. IPA フォントのダウ

ンロード環境や普及方法への提案（11件）、5. バグ報告（3件）、6. 表示技術上の問題（3件）であった。

3. 6. 2 Linux ディストリビュータからの反響

2008年8月時点で、日本のLinuxディストリビューションのうち、TurbolinuxとVMKNOPPIX、OpenSolaris、SUSE Linux Enterprise、OpenSuseがIPAフォントの標準バンドルを行っており、他のディストリビューションも標準バンドルの検討に入っている。

IPAフォントをOSに標準バンドルすることは、単独で利用するユーザーではなく、インターネットを介して情報を交換するあらゆるユーザーにとってのドキュメントの相互運用性の確保という意味で大変大きな意味を持つ。各ディストリビューションにとっても、他のOSやディストリビューションとの互換を担保できることは、自ディストリビューションの相互運用性向上となる。

しかし、一方でIPAフォントの現ライセンスでは標準バンドルできないといういくつかのディストリビュータからの意見も聞かれた。バンドルポリシーとして、OSIの承認をうけていることが前提となっているため、OSI承認のライセンスではないIPAフォントは搭載できないという見解である。

また、現在「一般利用者向けIPAフォント使用許諾契約書」は日本語のみであるが、英語ライセンスのニーズも大きい。これは、Linuxディストリビュータの多くが欧米を本拠地とする体制であるため、バンドルの承認過程も英語で行われるということに起因する。一般ユーザーであれば、日本語フォントをダウンロードすることは、日本語使用者であると解釈し日本語ライセンスのみで充分であると判断していたが、英語ライセンスへの対応を検討する必要がある。

3. 7. 考察

日本語フォントの情報通信基盤としての重要性とIPAフォント（Ver.002）の一般公開後の反響をふまえ、今後の日本語パブリックフォントの運用における課題を整理する。

情報通信基盤として日本語フォントを捉える際に重要なポイントは2点である。1点目は、情報交換性を保持することであり、そのためにはコンピュータ上で扱う

文字の定義である標準規格としての符号化文字集合に準拠した字形と文字コードを維持することである。2点目は、OSS やマルチ OS 環境で個別に調整をする必要が無いように相互運用性を高めることである。これら 2 点により、プラットフォームのデザインとして求められる適度な制約とルールとして創発的な価値創造を促すことが期待できる。

これまでの日本語フォントには、このような情報通信基盤となる日本語フォントが存在していなかった。従って、日本語パブリックフォントという新たなドメインの存在を広く公開し、オープンな環境で継続的にメンテナンスをすることが真の情報通信基盤としての役割であると考えられる。

以下に日本語パブリックフォントの運用における具体的な課題を示す。

(1) フォントメンテナンス

フォントを情報通信基盤として成立させるためには、メンテナンスなしに配布し続けることはできない。フォントのメンテナンスをどのように行うかが、公共性の高いパブリックフォントのテーマである。

主なメンテナンス要素として a) オープンモジュールとしてのフォントの調整、b) 規格の改定が上げられる。

a) オープンモジュールとしてのフォントの調整

図 4 に示した通り、フォントはオープン・アーキテクチャの 1 モジュールとして機能しており、フォントフォーマットというオープンな仕様により、他のモジュールとの調整を取っている。しかし、実際にはフォントフォーマットの仕様には多数の文字属性を記述する Feature テーブルが存在しており、日本語の文字処理において各 OS や各アプリケーションがどのテーブルを使用するかが明確に定義されておらず、そのためマルチプラットフォームを前提とした日本語フォントには潜在的な不具合が存在している。

これは、フォントフォーマット自体が米国企業を主体として制定されており、特に日本語環境に特化した処理方法を詳細にフォーマット化しているわけではないため、フォント開発側もアプリケーション開発側も個別の判断で使用する Feature テーブルを組み込んでしまっていることに起因する。

今後、フォント側が不具合に対応することも重要であるが、日本語環境に必要な Feature テーブル情報の共有を促進することで、各方面の開発コストを減らすことが期待できる。

b) 規格の改定

文字コード（符号化文字集合）の規格は、国の施策などとリンクしており、追加や改正がありうる。日本語パブリックフォントが情報通信基盤として早急に新しい規格への対応を行い、フォントメーカーの参考となる実装を行うことは、公共性の高いフォントとして重要な役割である。

同様にフォントフォーマットについても、将来的に変化が起り得る。既存のフォントを改訂したり、コンバートしたりすることを前提にしておくことで、日本語パブリックフォントの永続性を保つことができる。

近々のテーマとして、Unicode で規格化された Ideographic Variation Sequence (IVS) と Ideographic Variation Database (IVD) ²⁹への対応がある。

Unicode をはじめとする文字コード規格では、1つの文字に1つの固有な番号を付与し、文字の具体的な字形を規格票にまとめる上で1つの例示字形を示している。この例示字形はなんら規範的な役割はなく、包摂規準に沿った字形のどれをフォントに実装しても問題ないが、フォントのメカニズムとしては1フォントには1つの文字コードに1字形しか割り当てられないことから、地名、人名等に多く見られる同じ文字であるが異なる字形で表示したいという要望には充分応えられないという問題があった。

IVS と IVD の実装により、多くの字形を必要とする自治体などのニーズへ対応することが可能となる。

(2) 相互運用性と技術イノベーション促進

日本語パブリックフォントを利用することにより、異なる OS 間、異なるディスプレイ間での電子メディア交換において、同一のドキュメントが異なって見えてしまうという相互運用性問題を解消する。多くの人々がパブリックフォントを

²⁹ Unicode Home Page 「Ideographic Variation Database」：
<http://www.unicode.org/ivd/#reviewinstructions>

使うことで、さらに利用が促進され、より有効に利用されるというネットワーク外部性が働く。従って、今後日本語パブリックフォントの利用を促進することが、日本語環境における情報通信基盤の整備という意味で重要である。

また、OSSをはじめとするソフトウェアの技術イノベーションの促進において、日本語パブリックフォントを中心としたオープンな情報交換の場が必要である。

パブリックフォントを、異なる環境の多数のユーザーが使用することで実用的な問題が抽出され、フォントに問題があるのか、OS に問題があるのか、アプリケーションに問題があるのかなどのフィードバック機能が働き効率的な技術開発へとつながる。さらに、規格やデザインなどのユーザーニーズの集積や、フォントやその関連技術に関する知識の共有を促進することで、新しい技術開発要素を生み出すことが期待できる。

米国主導のソフトウェア産業において、日本の国際競争力を向上させるためには、日本語環境というネックをいち早く取り除くことが重要である。

(3) コミュニティ作り

上記の(1) フォントメンテナンス (2) 相互運用性と技術イノベーション促進を具体的に実行する環境として、フォント開発環境と、オープンな情報交換の場が必要である。

a) フォント開発環境

フォントは継続したメンテナンスを必要とする。商用フォントはビジネスとしてフォントメーカーが継続してメンテナンスしているが、特に日本語フォントは文字数が膨大なため、メンテナンス、開発継続には多大なる労力とコストが必要である。ソフトウェア産業の育成のためには、日本語フォント環境の充実が必須とされる中、これまでのように改変の必要に応じて調達するような方法ではなく、OSSと同様の形態で草の根的にフォント開発を継続できるコミュニティの育成が求められる。

日本語フォント開発環境として、商用フォントメーカーは高価なフォント開発ツールと自社内に独自に構築した自社専用の開発環境を使用しており、一方フリーで入手可能なフォント開発ツールは日本語フォントを効率的に開発するには機能が充分でない、という問題がある。

IPA では、パブリックフォントの継続開発にあたり OSS のフォントツール³⁰の採用を検討したが、日本語フォントを編集するための機能を満たしておらず、結果として独自に IPA 内にパブリックフォントの開発環境を整備し、その環境を提供することとした。

今後、IPA 内に構築した開発環境をオープンにフォント開発者が利用できるようにインターネット上に開放し、さらに、複数のフォント開発者が分散開発を行うことを前提とした開発手法の確立が必要と思われる。

b) パブリックフォント開発コミュニティ

高品質な日本語フォントを継続開発する環境を整備するとともに、OSS 手法を導入し、フォント開発技術者やレンダリングエンジンなどのフォント関連技術開発者の育成、研究推進を行う。

商用フォントは品質維持のため、限られたデザイナーと技術者により美的統一維持や機能面でのメンテナンスのノウハウを蓄積してきた。商用フォントに重要な要素は、他のフォントとの美的差別化やフォントを搭載する環境に合わせたユーザビリティの追求にある。

パブリックフォントは、商用フォントに置き換わるために存在するのではなく、情報通信基盤となる日本語表示を提供するとともに、日本語表示のベースとなる技術の継承やイノベーション促進に寄与する体制を築くことに意義がある。

従って、パブリックフォントを継続開発する環境を提供するとともに、フォントを維持継続するコミュニティを育成し、情報を共有できる環境作りが重要である。このコミュニティには、フォント開発技術者だけでなく、文字デザインを行うデザイナー、文字の字形統一を判断する専門家、フォント開発環境を構築する技術者、文字表示に関連する技術者など異なった知識を所有するメンバーを動員することで、日本語フォントのあるべき姿の実現を目指す。パブリックフォント開発コミュニティによる知識の集約により、フォントのみならず日本語使用を行うあらゆるソフトウェアの効率的開発を支援する。

³⁰ OSS のフォントツールには FontForge、TTX/Font Tools がある。

c) パブリックフォントユーザーコミュニティ

日本において、フォントのインストール方法やフォントのコア技術などフォントに関する情報をオープンに交換するためのコミュニティは存在していない。

フォント技術は、オープンスタンダードなフォントフォーマット以外は、フォントメーカー内部でのクローズドな情報か少数のフォントに関心を持つ個人の公開情報しかなかった。1つの要因としては、フォント開発は使用 OS や使用目的などフォントの個別のニーズに合わせて行われているため共有すべき情報が少なかったことが挙げられる。

日本語パブリックフォントについて、ユーザーコミュニティという場で情報を交換することにより、OSS 活動におけるフォント活用はもちろん一般利用者にとっても意義のある場となり、利用促進にもつながる。

今後、パブリックフォントユーザーコミュニティの Web サイトを立ち上げ、誰でもコミュニティへの参加が可能な形で情報交換を行う場の提供を行うことが望ましい。

3. 8 おわりに

本章により、日本語フォントが OSS をはじめとするソフトウェア産業において重要なオープンモジュールであり情報通信基盤であることが明確となった。

社会環境の変化として OSS の普及ならびに日本政府が OSS を推進し始めた当初を見た場合、オープンな日本語フォントがなかったため OSS における日本語環境整備にあたって日本語フォントがボトルネックとなっていた。この問題を解決するために IPA は IPA フォントを公開した。IPA フォントは、OSS やマルチ OS 環境で自由に利用可能なフォントとして普及しはじめている。

本章では、今後の技術と社会環境の変化に対応し、日本のイノベーションを加速させるための公共フォントとしての日本語パブリックフォントという新たなドメインの課題を提示した。日本語パブリックフォントが、情報交換性を保持するという制約と、相互運用性を高めるためのルール作りを行うための機能を持つことで、創発的な価値創造につながることを期待する。

今後は、実践例としてあげた IPA フォントに対して、本研究で提案した日本語パブリックフォントの課題を基にデザインした機能実装をおこない、その結果をフィ

ードバックすることで、再度問題の抽出、デザインの変更をおこなっていく。

そして、長期スパンでの日本語パブリックフォントの普及状況の観察やパブリックフォントコミュニティから発信される情報のつながりを観察することで、実社会と日本語パブリックフォントの関係を明らかにする。特に、パブリックフォント開発コミュニティにおいて議論される日本語環境に関する情報交換が、日本語フォントのみならず日本語使用を行うあらゆるソフトウェアの効率的開発を支援する機会となることを期待するとともに、日本語フォントのあるべき姿について提案することが今後の課題である。

また、IPA フォントが OSI 承認を前提としたライセンス形態を取るために、新たに日本語パブリックフォントライセンスを策定した。このライセンスについて、第四章で述べる。

第四章 日本語パブリックフォントライセンスの研究

4. 1 はじめに

本章では、第三章で提案した情報通信基盤を支えるデジタル財の1つであるオープンな「日本語パブリックフォント」を自由に流通させるために最適なライセンスについて、DREの視点に立脚して検討を行う。

4. 2 DREとは

4. 2. 1 デジタル財と著作権

インターネット上で交換可能な「デジタル財」は、時間や信号強度の連続である「アナログ財」と異なり、ビットの一連の流れにより符号化されたものであることから、適切な技術とコストにより、ほぼ完全な複製と再生が可能である（曾根原、2004）。

利用者にとって「デジタル財」は技術的には完全な複製が容易に可能なものであるが、「デジタル財」の情報流通最適化を考える場合には、その情報の著作権者の権利を保護しつつデジタル財の活用の活性化を計るという視点が重要となる。これは、我が国の著作権法第1条の「文化的所産の公正な利用に留意しつつ、著作者等の権利の保護を図り、もって文化の発展に寄与する」ことに相違ない。

これまで、「デジタル財」を含む「情報財」は、レコード、CD、DVDなどの有体物に体化されることで財貨性を持たせることができていた。しかし、ネットワークのブロードバンド化は、有体物の体化過程を経ることなく、デジタル流通を可能にした（林、2005：132）。著作者がその著作物の流通をコントロールするための法的な仕掛けが著作権制度であり、有体物の存在は流通プロセスの中のボトルネックを支配すること、たとえば書物という著作物における印刷という出版社の行為を支配すること、を可能としていた（名和、2004:163）。

しかし、デジタル流通ではこのボトルネックは消失してしまった。したがって、著作者の権利を保護するためには、新しい著作権制度作りが必要となった。

4. 2. 2 デジタル権利管理技術 (DRM)

新しい著作権制度作りの一つにデジタル権利管理（以下、DRM：Digital Rights Management）技術がある。DRMとは、著作物に権利管理情報（RMI：Right Management Information）を付加し、配信およびアクセス制御や使用・コピー制限等の著作物管理を行う技術である。しかし、DRMには、その費用を誰が負担するのか、デジタル時代における使い易さと権利保護のバランス、私的や教育的使用などの使用条件認証、購入履歴などのプライバシー保護、機器認証と他の機器での保存や視聴というシームレス権利流通などを始めとして多くの課題がある（沼田・福田、2009:29）。また、DRMを無効化する機器やプログラムも横行するようになった。その技術をさらに無効化する技術開発を行うなど「イタチごっこ」の状態が続くようになった。このようなことが横行することはコンテンツ産業全体のビジネスが成り立たないことになってしまうため、技術制限手段に対する不正行為を不正競争の類型として、平成11年度に不正競争防止法に追加される改正がなされた（松田、2009:23）。

このようにして DRMにより、商用のデジタル財の財貨性をコントロールすることで著作者の知的財産権を保護することは可能であろうが、一方で複製、改変、頒布が容易であるというデジタル財の利点を享受するためのしくみとは言い難い。

著作権法は、100年以上前に制定されたベルヌ条約を基礎としており、デジタル流通には適さない内容がある。その1つが著作物の利用に関して不許可（禁止）がデフォルトであるため、利用に関して明示的に表記されていないときには利用してはいけないことを意味することであり、もう1つに複製、改変、公衆送信などの行為について別個の禁止権が設定されており、すべての許諾がなければ違法となってしまうことである（野口・堀岡、2007:101）。

4. 2. 3 デジタル著作権表明 (DRE)

デジタル財に対する不適合への対処を制度設計から提案するのがデジタル著作権表明（以下、DRE：Digital Rights Expression）である。「権利者には、使用条件を予め明示する道、すなわち著作権表明の道を開いておくことこそ、制度設計の原点」（林、2008:68）という考え方である。

現在利用されている代表的な DRE には、Free Software Foundation³¹によって公開・維持されているプログラム向けライセンスの GNU Public License（以下、GPL）や、2001年にスタンフォード大学ロースクール教授の Lawrence Lessig らによって米国の非営利団体として設立されたクリエイティブ・コモンズ（Creative Commons）による Creative Commons Public License（以下、CCPL）などがある。これらの DRE に共通する利点は、あらかじめ著作者の権利とライセンス条件を明示することで、利用者が利用許諾を得るための非生産的な作業を省けることである。つまり、デジタル財の活用に重きを置いているのが DRE と捉えることができる。

商用のデジタル財では従来から使用許諾契約によって利用者への許諾の範囲を明示してきたが、原則は複製、改良、頒布できないものである。このような商用ソフトウェアに対する使用許諾契約による利用制限に対し、ソフトウェアの設計図にあたるソースコードを無償で公開し、誰でもそのソフトウェアの改良、再配布が行えるようにするライセンスとして作成されたのが GPL である。GPL は、コピーライト（著作権）に対し、「プログラムを実行し、コピーし、書き換えることが可能であり、また書き換えられたバージョンを流通させられるが、書き換えバージョンであってもいかなる独自の制限事項も追加してはならない」コピーレフト（Stallman、2003）という価値観を生み出した。GPL を発端に、オープンソフトウェアという DRE による開発モデルは、LinuxOS や WWW サーバの Apache などのさまざまな形のプログラムとして普及していった。これらのプログラムの普及には、そのライセンスの伝搬性があげられる（野口・堀岡、2007:104）。あるライセンスで公開されたプログラムの派生プログラムは、元のプログラムのライセンスと同じライセンスで公表することが義務付けられることで、利用者は新たな権利処理をする必要がなくなる。

さらに、オープンソフトウェア運動に触発されて、コンテンツの領域においても、同様に自由な共有に基づく文化が可能であるべきだ、として設立されたのがクリエイティブ・コモンズである（クリエイティブ・コモンズ・ジャパン事務局、2006:388）。Google の検索エンジンを利用した推計によると、2006年6月の段階で Web 上に 1

³¹ Free Software Foundation : <http://www.fsf.org/>

億 4000 万個のデジタルファイルが CCPL で公開されており（クリエイティブ・コモンズ・ジャパン事務局、2006:390）、ライセンスの伝搬性により拡大の一途をたどっている。著作権完全保護（full copyright）と著作権放棄（public domain）の単純な二項対立状態を分節する中間領域の設計（生貝他、2006:584）として、デジタル財のイノベーションを促進するネット社会の不均衡を是正するしくみといえる。

4. 2. 4 日本語パブリックフォントと DRE

これらの DRE により提供されるデジタル財を、本研究では「オープンなデジタル財」と定義する。オープンなデジタル財のうちオープンなプログラムには GPL、オープンなコンテンツには CCPL に代表されるデジタル財の発展を促進する DRE が存在するが、情報通信基盤を支えるオープンな「日本語パブリックフォント」に適合した DRE が存在しない。

情報通信基盤には、全国的なブロードバンド基盤、有線・無線のシームレスなアクセス環境というように物理的な基盤と、プライバシー保護、情報セキュリティ、電子商取引環境など IT の利用環境の整備（岡崎、2004:118）のようなソフトウェア的な基盤があるが、ソフトウェア的な基盤には文字を表示するためのフォントが必要不可欠である。つまりオープンな「日本語パブリックフォント」とは、情報通信基盤としてさまざまな IT 利用環境において相互運用性を持って利用できる日本語フォントというドメインを指している。

フォントは、その性格上プログラムとコンテンツの両方の要素を含んでいるが、加えてフォント特有の要素もあり、かつ日本語フォントには日本語特有の保護すべき要件を備えている。しかしながら、いままでにフォントを DRE の視点で捉え分析した研究は存在しなかった。

第三章で示したように、IPA フォント（Ver.002）ではライセンスについて課題が残った。開発者側、特に Linux ディストリビュータから、改変や修正した派生フォントの頒布ができるライセンスの要望があった。いくつかの Linux ディストリビュータでは、Linux ディストリビューションにソフトウェアを標準で利用できる状態でバンドルするポリシーをそのソフトウェアが OSS であることとしており、IPA フォント（Ver.002）はライセンス上 OSS とは認められないため、バンドルできないという問題である。

IPA フォントを日本のソフトウェア産業の活性化のための情報通信基盤として捉えた場合、相互運用性の確保は大きな課題であり、そのためには多くのユーザーが IPA フォントを入手し利用できる環境整備を必要とする。

この IPA フォントのライセンスをめぐる課題を踏まえて、情報通信基盤を支えるオープンな「日本語パブリックフォント」に適合した DRE の検討を行う。

4. 3 フォントの法的保護の現状と問題の所在

4. 3. 1 フォントの法的保護の現状

デジタル財となったフォントは、標準化・規格化されたことにより、だれもが簡単に複製し改変することが可能となった。活字、写植というアナログ時代でも複製は技術的には可能であったが、複製のためには膨大な時間と技術が必要であった。しかし、デジタル時代には、フォントを入手しそのまま販売する海賊版の他に、多少の変更を加えて新しいフォント（模倣フォント）として販売する行為が容易に行える。

フォントの知的財産権としては、日本法では、フォントはタイプフェイスのデザイン（字形の輪郭線）を高次の数式で「表現（描画）したプログラム」及び文字属性データベースから構成されており、プログラムの著作権とデータベースの著作権で保護される可能性をもっている。

プログラムの著作権に関連する判例として、株式会社モリサワを原告とする著作権侵害に基づく差止等請求事件（大阪地方裁判所 平成15年（ワ）第2552号）がある。パーソナルコンピュータ用フォントを製造・販売している原告が、被告株式会社ディー・ディー・テックがそのプログラムを違法に複製して原告の著作権（複製権）を侵害したと主張し、裁判所は当該プログラムの使用を差止め、損害賠償金を支払わせる判決を下した（裁判所ウェブサイト A）。この判例は、いわゆる海賊版に対する不正競争防止法第2条第1項所定の不正競争行為に該当する行為（商品の形態を模倣したデッドコピー）に基づく判例であり、プログラムとしての著作権侵害を認めた判例とは言い難く、フォントがプログラムとして著作権法で保護されるかという議論とは異なる。しかし、不正競争防止という視点でデッドコピーから保護することが可能であると言える。フォントを作成するプログラムについては、プログラムとして著作権法で保護される可能性は高いが、字形の輪郭線を高次の数式

で「表現（描画）したプログラム」の著作権については、判例としては明らかになっていない。

データベースの著作権については、日本法で規定上は、選択、配列（体系的構成）と言った点において創作性が保護の根拠になっているが、実際にはこれらの点において創作性が認められないものが多い（相山、2005:123）とされている。フォント・プログラムの中の文字属性データベースに創造性が認められるか否かは明らかになっていない。たとえば A フォントの文字属性データベースを使って B フォントを作成した場合に、著作権侵害になるかどうかは判例が無いため明らかではない。

一方、タイプフェイスの保護に対する問題がある。日本タイポグラフィ協会によると、タイプフェイスとは「言語表記を主目的に、記録や表示など組み使用を前提として、統一コンセプトに基づいて制作されたひと揃いの文字書体」である（日本タイポグラフィ協会 B）。この定義によると、フォントは、タイプフェイスを具体的な記録や表示、印刷などに利用できるようにしたハードウェア（アナログフォント）、ソフトウェア（デジタルフォント）である。

日本法において、タイプフェイスは著作権法や意匠法で保護されていない状況にある。

著作権上の判例として、平成 12 年 9 月 7 日に最高裁判所が上告棄却した「著作権侵害差止等請求本訴、同反訴事件」³²がある。本件は印刷用書体の著作権侵害を争うものであったが、上告人書体が著作物とはいえないという判決が下されている。その理由として、「印刷用書体がここにいう著作物に該当するというためには、それが従来の印刷用書体に比して顕著な特徴を有するといった独創性を備えることが必要であり、かつ、それ自体が美術鑑賞の対象となり得る美的特性を備えていなければならないと解するのが相当である。」とする一方、「印刷用書体は、文字の有する情報伝達機能を発揮する必要があるために、必然的にその形態には一定の制約を受けるものであるところ、これが一般的に著作物として保護されるものとする、著作権の成立に審査及び登録を要せず、著作権の対外的な表示も要求しない我が国の著作権制度の下においては、わずかな差異を有する無数の印刷用書体について著作

³² 大阪地裁平成 5 年(ワ)2580 号、9208 号平成 9 年 6 月 24 日判決（共棄却）；大阪高裁平成 9 年(ネ)1927 号平成 10 年 7 月 17 日判決（棄却）；最高裁平成 10 年(受)第 332 号平成 12 年 9 月 7 日判決（棄却）

権が成立することとなり、権利関係が複雑となり、混乱を招くことが予想される。」としている（裁判所ウェブサイト B）。

タイプフェイスに対する著作権法による保護は完全に否定されるものではないが、顕著な独創性および美的創作性そなえていることが条件となり、情報伝達手段として実用的なタイプフェイスの場合には、その保護の対象となることは難しい。

また工業製品の意匠を目的として作られた意匠法では、形状のある物品を保護対象としており（特許庁総務部総務課制度改正審議室、2006：13）、タイプフェイスは物品ではないため保護されない。

中塚（2008）によると、海外のタイプフェイスの保護の実態（表 7）としては、米国はタイプフェイス及びフォントを意匠特許の保護対象としているが、著作権法では保護されていない。欧州では、タイプフェイスが意匠権の保護対象となることが明文化されているが、有体物（活字や写植機の文字盤）が対象でありデジタルデータは含まれない。米国ではタイプフェイスを作成するフォントソフトプログラムは著作権保護の対象となっている、欧州では、フォント・プログラムは著作権で保護されており、保護期間の短い意匠特許および意匠権よりも、著作権登録による保護が利用されている。韓国では、2004年に意匠法をデザイン保護法と改称し、タイプフェイスの保護を明文化しているが、その効力はタイピング・組版又は印刷等の通常の過程で書体を使用する場合及び書体の使用により生産された成果物には及ばないという効力制限規定が置かれている。

表 7 各国でのフォントおよびタイプフェイスの知的財産権の状況

(中塚(2008)等から筆者作成)

		意匠権	著作権
日本	フォント	×意匠権	△(フォント作成プログラムの著作権)
	タイプフェイス	×意匠権	×
米国	フォント	○意匠特許権	△(フォント作成プログラムの著作権)
	タイプフェイス	○意匠特許権	×
欧州	フォント	×意匠権	○プログラムの著作権
	タイプフェイス	△意匠権(*注)	×
韓国	フォント	○デザイン保護法	- (未調査)
	タイプフェイス	○デザイン保護法	- (未調査)

(注) 有体物(活字や写植機の文字盤)が対象でありデジタルデータは含まれない。

このように海外においては、多少なりともタイプフェイスに対する法的保護が進んでおり、日本のデザイナーやフォントメーカーから法的保護を望む声は高まっている。

日本タイポグラフィ協会は、上記著作権に関する最高裁判所判決に対し、「法令解釈の間違い、あるいは他の著作物には要求しない、タイプフェイスのみへの不当な要求」という声明とともに「タイプフェイスは我々の文化を支える著作物の一つ」として継続したタイプフェイス保護の活動を行っている。

一方、フォントベンダーの働きかけにより、内閣官房知的財産戦略本部(2007)の「知的財産推進計画2007」において、知的財産権制度の強化の1つとしてタイプフェイスの保護を強化することが掲げられた。しかし、現在のところ新しい法的保護制度は考えられていない。中塚(2008)によると、有識者委員会において、情報伝達の媒介である文字を基本としたタイプフェイスに何らかの法的保護を与えることは、産業界のみならず国民の生活にまで影響を及ぼす可能性があり、法的保護については具体的な問題に沿って慎重に検討する必要があるため、新たな法的保護の制定は時期尚早とされている。

このように、フォントおよびタイプフェイスは著作権法上必ずしも保護されている状況にはないが、現状は、日本のフォント業界における商習慣として、個別契約(使用許諾契約書)によりフォントおよびタイプフェイスの使用制限を設け、民法

上の債権債務による保護を行う方式が定着している。

一方、準拠法の観点で考察すれば、特に米国と日本の法律の間で最も整合性が取れていないのが著作権者人格権（著作権法 59 条）である。日本の法律下では著作権者人格権を他人に譲渡や放棄できないと解されているが、アメリカ合衆国著作権法は著作権者人格権を扱う規定を設けていない。著作権者人格権とは、著作物を作った人物（著作権者）の権利を保証するもので、他人が著作権者に無断で著作物を公表したり、著作権者の名誉を損害する方法で著作物を利用したりすることを禁止している。

また、著作権者人格権には、著作権者の許可なく表現を勝手に変えてはならないという同一性保持権（著作権法 20 条 1 項）が含まれる。この同一性保持権が GPL やクリエイティブ・コモンズと整合しない場合がある。クリエイティブ・コモンズでは、日本法準拠のローカライズ版作成時に「第 2 条 著作権等に対する制限」において、著作権者人格権の不行使を選択せず著作権者の名誉を傷つけることなく改変することを認めるライセンスになっている。一方、米国生まれの GPL では著作権者人格権に関しては触れられていない。

GPL もクリエイティブ・コモンズも、著作権者が「自分はこのように使ってほしい」という意思の主体的宣言であり、利用者はその宣言を承諾（使用許諾を承諾）することで利用できる。つまり、著作権者の表明として許諾するものであり、日本では著作権者人格権や著作権者以外の公衆送信行為を規制する公衆送信権（著作権法 23 条 1 項）などとの整合性も考えれば、著作権法に依拠するのではなく、著作権者と利用者間における使用許諾の契約であるとして捉えられる。

フォントの場合、本章で考察したとおり、著作権に関しては明らかになっていない。従って、GPL と同様の考え方をとり、各国の法的根拠に基づくものでなく、著作権者による使用許諾表明であり利用者が契約するという考え方で日本語パブリックフォントライセンスを検討する。

4. 3. 2 問題の所在

本研究は、フォントの法的保護の強化を提案するものではなく、フォントをデジタル財として自由に流通させることで相互運用性を高め、情報通信基盤として活用するための手段を論ずるものである。これまでの先行研究では、フォントの保護に関する側面ばかりが追求されてきたが、このままでは、維持・メンテナンスに多大

なコストが必要となる日本語フォントの健全な発展にも支障をきたしていく可能性が高い。また、現状の商習慣である購入時の個別契約（民法上の債権債務）では、原則は複製、改良、頒布は契約違反となり、さらに、開発者(社)またはフォント毎でその使用許諾範囲が異なるため、ユーザーは使用許諾をフォント毎に解釈し、その範囲で使用すると言う判断を必要とし、ユーザーにとって必ずしも使い勝手の良い物ではない。ユーザーが安心して利用可能な形態であることが、フォントをデジタル財として流通し活用するためのポイントの一つである。

これまで、本研究がとりあげる IPA フォント以外に国内外でパブリックフォントといえるフォントは存在しなかった。日本では、商用以外ではフリーフォントと呼ばれる無償フォントが配布されているが、権利関係が明確でかつ高品質なフリーフォントが存在しなかった。OSSの世界では、2003年にLinuxディストリビューションのほぼすべてに収録されていた東風（こち）フォントがあったが、権利侵害の問題が取り上げられ、フォント制作活動の終了に至っている。東風フォントが、商用のビットマップフォントを元として修正を加えアウトライン化したフォントであったことから、著作権侵害の可能性を指摘されたが、日本法において権利侵害にあたるかは明確にされなかった（池田・沼田・兼子、2009）。

東風フォント問題からも、オープンでパブリックなフォントを継続して使用し維持・メンテナンスするためには、法的な保護に頼るのではなく、明確なパブリックフォントとしての使用許諾の表明が必要であると考えられる。

本研究では、フォントの保護の側面、デジタル財としての流通の側面に対して、パブリックフォントという概念をとらえ、フォントに適したDRE導入を検討することにより、相互運用性に富んだ情報通信基盤の確立へ寄与するものである。

4. 4 フォント向けライセンス

4. 4. 1 フォント向けライセンスの現状

商用フォント、IPAフォント（Ver.002）ならびにOSSを中心に利用されているフリーフォント向けとされているライセンスを取り上げ、その特徴をあげる（表8）。

（1）商用フォントライセンス（使用許諾契約書）

商用フォントライセンスでは、複製、改変、再頒布、フォントから文字情報を取

り出してフォントや二次成果物を作成する、などの行為は当然禁止事項とされており、その上で使用の範囲の制限が存在する。特に、近年のデジタル化、ネットワーク化の進展は、デジタルフォントの使用用途を拡大したが、1 ライセンスの適用範囲が不明瞭となる問題が生じている。以下に商用フォントが制限を行っている項目を挙げる。

- (1) 使用環境として提示した OS にのみインストールができる。
- (2) 1 ライセンスで許諾する単位を「コンピュータ 1 台」として複数の CPU が搭載されている場合、複数の OS がインストールされている場合、複数の OS が同時起動している場合も 1 台と見なす。
- (3) 映像コンテンツまたはデジタルコンテンツにおいてフォントとして機能する形態で使用する場合は別途契約が必要である。
- (4) ネットワークサーバー上にインストールしたフォントをクライアントのコンピュータで使用することは禁止する。ネットワークサーバーへのフォントのインストールを禁じる場合もある。
- (5) ドキュメントにフォントをエンベッドすることを禁じる場合もある。

(2) IPA フォント (Ver.002) ライセンス「一般利用者向け IPA フォント使用許諾契約書」

IPA フォント (Ver.002) ライセンスは、IPA フォント (Ver.001) が IPA の支援した「オープンソフトウェア活用基盤整備事業」で開発されたソフトウェアに同梱する場合のみの使用許諾に限定したものであった事に対し、広く多くのユーザーに使用を許諾することを前提として作成された。

以下が主な特徴である。

- (1) 一般利用者にとっての、デジタルコンテンツでのフォント利用に対して最大の自由度を付与する。(商用フォントにおける使用範囲の制限を排除する。)
- (2) 個人利用において、改変、修正を自由にする。
- (3) 提供する IPA フォントの頒布は商用・非商用を問わず自由に行えるが、改変や修正した派生フォントの再頒布は不可とする。

これまでの商用フォントライセンスにおけるサーバ使用の不許可などのデジタル利用特有の使用用途の拡大に範囲については、曖昧さを排除して利用を許可した。

但し、派生フォントの再頒布については、日本語の真のパブリックフォントというドメインを築き、一方でフォント市場と文化を維持しつつ、IPA フォントの品質を保つための十分な環境整備が必要であるという考えのもと、IPA フォント (Ver.002) ライセンスでは不可とした。

(3) GPL v2 + font exception (Free Software Foundation, Inc., 2009)

GPL は、FSF によって公開・維持されているソフトウェア向けライセンスである。GPL でライセンスされたソフトウェアは、利用者に対してソフトウェアの自由な改変・頒布を行うことを許諾する一方で、改変したソースコードを公開し、同様の条件で利用許諾を行うことを求めている。ただし、フォントに関しては、font exception を追加することで、例外的にフォントを使って作成されたドキュメントについては GPL が適用されないことを明記することができる。

font exception はフォントがフォント・プログラムとして独立して頒布されるだけでなく、フォントの一部または全部がフォントを使用したドキュメントにエンベッドできるため「プログラムのリンク」という行為として捉えられるが、そのドキュメントに GPL を適用することは利用上不都合であるという特殊性に考慮した例外事項である。

(4) Creative Commons Public License (CCPL) (Creative Commons, 2009)

クリエイティブ・コモンズによる CCPL は完全な著作権保持と完全な著作権放棄の中間を埋める役割を目指すライセンスであり、さまざまなコンテンツを対象としている。

CCPL は、利用許諾条件の表明を行う DRE の概念を導入してコンテンツを流動化させようとするものであり、ライセンスの条件を 4 つのコンポーネントに分割し、それらの組み合わせによって権利者の必要に応じた柔軟な条件を指定できる。ライセンスのコンポーネントは「表示 (Attribution)」「非営利 (Noncommercial)」「改変禁止 (No Derivative Works)」「継承 (Share Alike)」の 4 つから構成されており、これらを組み合わせることで「表示」「表示-非営利」「表示-非営利-改変禁止」「表示-改変禁止」「表示-非営利-継承」「表示-継承」の 6 種類のライセンスの形態が利

用可能となっている。

フォントライセンスとしては、「表示」または「表示-継承」のライセンスが適合しているとされる。

「表示」は作り手の名前を適切に表示すること、「継承」は作り手と同じライセンスで発表することであり、他のフォントライセンスと比較すると言葉足らずな印象をうける。

(5) Bitstream Vera ライセンス (Bitstream, Inc., 2003)

Bitstream Vera ライセンスは、オープンソースフォントとして Bitstream Inc. が提供している Bitstream Vera フォントに付与されているライセンスである。Bitstream Inc.は、フォントの他にブラウザや出版関連のソフトウェアを開発する企業であり、1000 書体以上のフォントをリリースしている。

Bitstream Vera ライセンスは 2003 年に作成され、このライセンスで頒布している Bitstream Vera フォントは欧文フォント 10 書体であり、派生フォントは認めているが、フォント単独での販売と、派生フォントのフォント名には「Bitstream」と「Vera」という語を使用してはならないという制限がある。また、ライセンス内に Bitstream Vera フォントを対象とした表現があるため、Bitstream Vera フォントおよびその派生フォントしか対象とならないが、フリーフォントライセンスの基礎となるライセンスである。

(6) SIL Open Font License (OFL) ver.1.1 (2007 年作成) (SIL、2007)

OFL は SIL International (以下 SIL) が 2007 年に作成したフォントに特化したコピーレフトのオープンソースライセンスである。

SIL は、少数言語の保存とともに、世界中の様々な言語利用者が技術の恩恵を受けることが出来るように言語に関する研究開発や教育を支援する人道団体である。SIL はその活動の一部として、さまざまな文字・言語に対応した SIL フォントを公開している。

OFL は、フォント開発のコミュニティが世界レベルで育成されることを目標としており、フォントの自由な改変・頒布を許可している。OFL の特徴は、以下である。

1) 販売に対する制限

オリジナルおよび派生フォントにおいて、フォントのみを単独あるいはフォント集に含めて販売することを禁止するが、プログラムに同梱する限りは商用利用も可能である。

2) 派生フォントのフォント名に対する制限

オリジナルを改変した派生フォントは、オリジナルで予約されたフォント名 (Reserved Font Names) をその著作権所有者の許可なく使用できない。「予約されたフォント名」とは、予約されていることが著作権の宣言に続けて示されている名前のことである。

3) OFL の派生例外

オリジナルおよび派生フォントを使って作成したドキュメントには OFL は適用されない。

表 8 フォントライセンスの許諾内容

	商用 フォント	IPA フォント (Ver.002)	GPL v2 + font exception	CCPL	Bitstream Vera	OFL
ライセンス の適用フォ ント	特定	特定	不特定	不特定	特定	不特定
利用範囲の 制限	あり	なし	なし	なし	なし	なし
再頒布の自 由	×	OK	OK	OK	OK*注	OK*注
派生フォ ント作成	×	OK	OK	OK	OK	OK
派生フォ ントの再頒 布	-	NG	OK	OK	OK*注	OK*注
派生フォ ント名の制 限	-	-	なし	なし	あり	あり
派生フォ ントへのライ センス適用	-	-	あり	あり	あり	あり
ドキュメ ントへのライ センス適用	一部あり	なし	なし	定義なし	定義なし	なし

*注 フォント単独での販売は禁止。
ライセンスとしてオープンとはいえない内容や、適用が限定的な内容については、背景をグレーとした。

4. 4. 2 フォント向けライセンスの考察

ライセンスという観点では、フォントライセンスは、プログラムとコンテンツとは異なる配慮を必要とする。それは、フォントはプログラムやコンテンツのようにそれ自体が単独で活用されることはなく、OS やアプリケーションやコンテンツと連携することで初めて活用されるためである。

フォントの活用を具体的に示すと、OS やアプリケーションにおけるユーザーインターフェース部の文字表示、メール、オフィスソフト、電子書籍などのドキュメントにおける文字表示、画像や映像などのコンテンツの中に静的に埋めこまれる文字画像としての使用などである。

文字画像としてフォントとコンテンツが切り離されない限りは、OS やアプリケーションやコンテンツは「フォント名」により、フォントとの関係を維持しつづけることになる。たとえば、Aさんがオフィスドキュメントの本文に a フォントを指定し、そのファイルを B さんに渡し、Bさんのパソコン上で見た場合には本文は a フォントで表示されることが期待されている。これは、フォントがファイルに組み込まれているのではなく、フォント名がファイルの中に書かれており、Bさんのパソコンに a フォントと全く同一のフォント名のフォントが組み込まれていれば、そのフォントに収納された文字が表示されることを意味している。Bさんのパソコンに a フォントがなかったり、a フォント名を名乗る別のフォントが入っていたりすると、Aさんが送ったつもりで文字表示とは異なる文字が表示されることになる。

過去に作成された様々なプログラムやコンテンツは「フォント名」により、表示される文字の品質を担保していることになる。従って、フォント向けライセンスを考える上で、派生フォントの「フォント名」の取り扱いへの配慮が重要となる。

日本語パブリックフォントのライセンスとして DRE の視点で考察していくことの重要性は「4. 3 フォントの法的保護の現状と問題の所在」で見てきたが、DRE の代表的なライセンスであるプログラム向けの GPL やコンテンツ向けの CCPL では、「フォント名」の定義はできない。

フォント向けライセンスである OFL や Bitstream Vera ライセンスは派生フォントを許容してはいるが、その場合は名称を変更しなければならないと定めている。

「フォント名」の取り扱いへの配慮としては、ユーザーの混乱をさけるという視点が重要である。派生フォント間には整合性はなく品質も異なるため、同一名称に

するとインストールなどの運用面や、既存プログラムやコンテンツでの表現面で混乱がおこる。

さらに、フォントはタイポグラフィという創作性に対する主張の中で、派生フォントはオリジナルフォントとは美的統一維持の観点から保証できないため、オリジナルフォントに使われているオリジナルフォントを想起させる「フォント名」の一部を派生フォント名に使用することを許諾しないという考え方が存在する。

次節で日本語パブリックフォントライセンスを検討するにあたり「フォント名」に対する制限は必要不可欠である。また、Bitstream Vera ライセンスは、Bitstream Vera という特定のフォントのためのライセンスであるため DRE として幅広く適用可能であるライセンスとはいえない。オープンなフォントの DRE としては、不特定フォントへ適用可能ライセンスとし、派生フォントへの制限範囲をフォント特有の問題に絞り込んでいく必要がある。このような視点に立てば、既存ライセンスでは不十分であり、日本語パブリックフォントに適したライセンスはこれまで存在していなかったと言える。

4. 5 日本語パブリックフォントライセンスの確立

4. 5. 1 日本語パブリックフォントライセンスのための要件

日本語パブリックフォントとは何かという定義を行い、日本語パブリックフォントライセンスのための要件を整理する。

日本語パブリックフォントとは、それを利用する環境（OS、アプリケーション）が異なっても所望の文字を入力し日本語文章として見やすい形に組版されて出力（表示）できるという技術的な側面、すなわち、フォーマットと符号化文字集合が国際的に認められた標準に準拠しているフォントである。

同時に、利用者への許諾範囲を DRE によって明示することで、制限を極力なくし、自由な共有と開発が行えるような環境を構築できるようなフォントである。ただし、自由の中に「公共性の維持」という役割がある。このオープン性と公共性を併せ持った日本語フォントを本研究では「日本語パブリックフォント」と定義する。

日本語パブリックフォントの頒布の目的は、日本語情報通信基盤の整備とともにソフトウェア産業の育成にある。誰でも、どこでも、どのような環境でも使用できる高品位な日本語フォントは、日本語の文書交換を行う上での情報通信基盤である。

日本語パブリックフォントを利用することにより、異なる OS 間、異なるディスプレイビューション間の電子メディア交換において、同一のドキュメントが異なって見えてしまうという相互運用性問題を解消する。多くの人が日本語パブリックフォントを使うことで、さらに利用が促進され、より有効に利用されるというネットワーク外部性が働く。従って、日本語パブリックフォントの利用が促進されるようなライセンスが求められる。

また、ソフトウェア開発環境において、自由に利用可能な日本語フォントを活用することで、イノベーションの向上が期待される。日本語パブリックフォントはフォント産業を圧迫することを意図しておらず、ソフトウェア産業のイノベーションにより日本語フォントの利用シーンが増え、新たな日本語フォントニーズを生み出すことによりフォント産業も活性化するという正のスパイラルを期待するものである。

日本語パブリックフォントライセンスとして、OFL のようなフォントライセンスとの明らかな違いは、日本語フォントの特殊性を考慮する必要性にある。

日本語フォントは、欧米で用いられるラテン文字等とは異なり、多くの文字を収納しており、特に表意文字である漢字は、わずかなデザインの変更が文字の意味を大きく変えてしまい、ユーザーに混乱をもたらす可能性を持っている。1つの文字コードに割り当てられる漢字についても、同じ文字から派生した多数の異体字を持ち、人によってそのどれを用いるかの嗜好が異なる場合がある。さらに、間違った字の無意識な使用からの保護も必要である。

これまで、わずかな字形違いの文字の存在が情報交換上の事故やトラブルを多発させてきた。1字でも期待している字形が表示できなかつたり、間違った文字が表示されてしまつたりすると、システム全体の信頼性が失墜する。藤幡（2009:199）は、文字表示に何らかのトラブルがあると、文字により対象の描画を通して対象を理解する行為が、途端に視覚的体験へと「退行」してしまう、という。表示される文字が、人の情報理解の過程へと影響を与えてしまう、ということもできる。

情報通信基盤となるフォントにおいて、安易な字形変更を認めることは、情報の混乱を引き起こすだけにとどまらず、ユーザーおよび開発者の貴重な時間を膨大に奪い、情報通信基盤そのものの信頼性を損ねてしまう。ただ、純粹に利用の自由だけを追求している OFL 等の他のライセンスでは、この課題を解決することはでき

ない。

日本語パブリックフォントは、情報通信基盤の確立のために相互運用性の保持を目的とする。多くの環境でパブリックフォントを自由に利用できるようにすることが第1の責務である。その為に、オープンソースが定義する自由のポリシーを満たし、かつ、日本語の情報通信基盤としてオリジナルフォントが保証する文字への信頼性を確保することがフォントライセンスとしての要件としてあげられる。

4. 5. 2 IPA フォントライセンスの検討

日本語パブリックフォントライセンスとして、IPA フォントライセンスを検討するにあたり、「IPA フォント (Ver.002) ライセンス」から変更すべきポイントを整理する。

1つは、IPA フォントを普及し相互運用性を確保するために、多くのユーザーが入手しやすいものであることがあげられる。このために、以下に述べるように OSI (Open Source Initiative) 承認をうけることを検討する。

もう1つは、派生フォントを頒布可能とすることである。ただし、先に述べた通り派生フォントの制限事項として、「フォント名」の制限を設けることと、オリジナルフォントが保証する文字への信頼性を確保する手段を設けることが重要となる。

この2つのポイントを満たすためのライセンス内容を以下で詳細に検討する。

(1) OSI 承認

IPA フォントの普及には、IPA フォントライセンスを OSI 承認ライセンスとすることが挙げられる。相互運用性の確保のためにフォントを普及させるには、ユーザーが個別にフォントを入手するよりも OS レベルでの提供ことが効率的である。一般ユーザーにとって、フォントをインストールする作業は手間であるし、多くのユーザーは無意識に OS やアプリケーションが提供しているデフォルトのフォントを利用している。従って Linux ディストリビューションにバンドルできるライセンス体系の整備は相互運用性の確保に有効である。商用の OS やアプリケーションでのバンドルは各 OS やアプリケーションのディストリビュータの判断にゆだねられるが、Linux ディストリビューションにおいては、主流ディストリビュータである

Debian³³や Fedora³⁴などがソフトウェアのバンドルポリシーをそのソフトウェアが OSS であること、すなわち OSI 承認されたライセンスであること、としている。Linux ディストリビューションは商用・非商用を合わせても 30 種類以上あり、ベースディストリビューション (Debian、Fedora など) から展開した派生ディストリビューションの多くは、ベースディストリビューションのバンドルポリシーを踏襲している。

OSI³⁵とは、OSS の促進を目的として 1998 年設立された非営利組織である。OSI では "The Open Source Definition(以下 OSD)"によりオープンソースライセンスの基準を設けており、OSI では OSI 承認されたライセンスで頒布される場合を OSS としている。

これまで、フォントのためのライセンスで OSI 承認されたものは存在しなかった。

OSD では、表 9 に列記した 10 項目が OSS の基準とされている (Opensource.org、2006)。OSI 承認ライセンスとなるためには、OSD の 10 項目をすべて満たす必要がある。以下に IPA フォントライセンスの OSD 対応方針を整理する。

OSD のうち、「1. 再頒布の自由」については、再頒布は商用・非商用を問わず自由とする。

「2. ソースコードが入手可能」については、フォントにおいてはソースコードという概念が存在しないことを明確にしておく必要がある。多くのプログラムは、プログラミング言語で書かれたソースコードと、ソースコードをコンパイルしてできるコンピュータが実行可能なバイナリコード (オブジェクトコードとも言う) で構成されている。通常、一般ユーザーがプログラムとして使用しているのは、このバ

³³ Debian JP Project : <http://www.debian.or.jp/>

³⁴ Fedora Project : <http://fedoraproject.org/ja/>

³⁵ OSI が定義するオープンソースライセンスの必須条項 (OSD)は、複数のオープンソースコミュニティを交えた議論を経て策定されたものであり、OSD がオープンソースの定義であるとの共通認識が得られている。オープンソースという言葉がよく使われるようになってから、単にソースコードを公開しているだけで「オープンソースソフトウェアである」と自称するケースが目につくようになったが、このような誤用を是正することを目的として、OSI が審査して認定したライセンスには「OSI 認証」マークを使用して一般の利用者にも識別できるようにしている。

(<http://ossipedia.ipa.go.jp/kb/OSI> がオープンソースライセンスを認定しているのはなぜですか? から引用)

イナリコードである。ソースコードを入手可能にするということは、そのプログラムの設計書を頒布することであり、派生プログラムの開発者が効率的にプログラムの改変を行うことができる環境を提示することを意味する。一般的に、バイナリコードを改変することは困難であり、開発者がプログラムを変更しやすい形態としてはソースコードをなんらかの形で入手可能にしておくことが要求されている。しかし、フォントの場合は、フォントファイル自体がバイナリコードの性質（そのままコンピュータにインストールして実行できる）とソースコードの性質（改変するために必要なもの）を保有しており、プログラミング言語で書かれたソースコードというものは存在しない。現に既存のフォント開発プログラムでは、フォントファイルを入力すると、フォントに含まれるすべてのデータを読み込むことができ、改変を加えた後に派生フォントを出力することが可能である。フォントは、上述したソースとバイナリという関係を持つプログラムとは異なるソフトウェアであると言える。従って、フォントを頒布すること自体がすでに「ソースコードが入手可能」であることを満たしていると解釈できるため、本項目に対しては、特別な配慮をしない。ただし、効率的な改変のために、派生フォントを作成する過程でフォント開発プログラムによって作成されるフォントファイル以外の追加のファイルがあり、派生フォントをさらに加工するにあたって利用できる場合は、そのファイルも同時に入手可能とすることを明記する。

「5. 個人やグループに対する差別の禁止」、「6. 利用する分野に対する差別の禁止」、「8. 特定製品でのみ有効なライセンスの禁止」、「9. 他のソフトウェアを制限するライセンスの禁止」については、特別な制限は設定しない。

その他の項目については、派生フォントの制限事項として次項で検討をおこなう。

（2）派生フォントの頒布とその制限

OSD10項目をすべて満たし、かつ、オリジナルフォントが保証する文字への信頼性を確保するためには、オープンなフォントとして派生フォントの頒布を許可するが、日本語パブリックフォントとしてユーザーの混乱や利用における不具合を生み出さないための制限が必要となる。

「4. 4. 2 フォント向けライセンスの考察」で述べたように、ユーザーの混乱をさけるためには「フォント名」の取り扱いへの配慮が必要となる。したがって、

派生フォントのフォント名には、オリジナルフォントと同一の名称、またはこれを
含む名称を使用してはならないこととする。これは OSD の「4. 作者によるソー
スコードの完全性 (integrity) が認められる」という定義の下、派生ソフトウェア
に元のソフトウェアとは異なる名前やバージョン番号をつけるようことを義務付け
ることができていることから、OSD に抵触することにならない。

また、日本語フォントとしての文字の信頼性の確保と情報通信基盤としての位置
付けを考慮し、以下の 4 つの制限を設けることとした。

- (1) 利用者が、その意志により派生フォントをオリジナルのフォントに戻せる方
法を提供しなければならない。(表 9: OSD4 項目に対応)
- (2) 派生フォントには、それをさらに改変するために必要となる十分な情報を添
付しなければならない。(表 9: OSD2 項目、4 項目に対応)
- (3) 派生フォントにも、同じ IPA フォントライセンスを継承しなければならない。
(表 9: OSD3 項目、7 項目に対応)
- (4) 派生フォントは、Web 等のだれもがアクセスできる方法により一般公開しな
くてはならない。(表 9: OSD2 項目、10 項目に対応)

この 4 つの制限により、本来の日本語パブリックフォントが目的とする情報通信
基盤整備におけるオープン性と相互運用性、さらには複雑な日本語の漢字文化がも
たらす混乱を最小限に食い止めるフォントに対する信頼性の確保を図るものである。

表 9 OSD10 項目と IPA フォントライセンスの対応

OSD 10 項目	IPA フォントライセンス
1. 再頒布の自由	再頒布は商用・非商用を問わず自由とする。
2. ソースコードが入 手可能	フォントは、フォントファイル自体がバイナリの性質(そ のままコンピュータにインストールできる) とソースの 性質(改変するために必要なもの) を保有しており、ソ ースコードというものは存在しない。 派生フォントの制限(2): 派生フォントには、それをさら に改変するために必要となる十分な情報を添付しなけれ ばならない。 派生フォントの制限(4): 派生フォントは、Web 等のだ れもがアクセスできる方法により一般公開しなくてはな らない。
3. 派生ソフトウェア の許可	派生フォントの作成は許可するが頒布には制限事項を設 ける。 派生フォントの制限(3): 派生フォントにも、同じ IPA フ ォントライセンスを継承しなければならない。

4. 作者によるソースコードの完全性 (integrity) が認められる	派生フォントの制限(1)：利用者が、その意志により派生フォントをオリジナルのフォントに戻せる方法を提供しなければならない。 派生フォントの制限(2)：派生フォントには、それをさらに改変するために必要となる十分な情報を添付しなければならない。
5. 個人やグループに対する差別の禁止	制限は特になし
6. 利用する分野に対する差別の禁止	制限は特になし
7. 追加ライセンスの禁止	派生フォントの制限(3)：派生フォントにも、同じ IPA フォントライセンスを継承しなければならない。
8. 特定製品でのみ有効なライセンスの禁止	制限は特になし
9. 他のソフトウェアを制限するライセンスの禁止	制限は特になし
10. ライセンスは技術中立的でなければならない	派生フォントの制限(4)：派生フォントは、Web 等のだれもがアクセスできる方法により一般公開しなくてはならない。

4. 5. 3 OSI 承認プロセス

「4. 5. 2 IPA フォントライセンスの検討」で整理した要件を満たす「IPA フォントライセンス v1.0³⁶」を作成し、このライセンスを英訳した法律的視点から同一内容であるとみなせる英語版ライセンスが OSI 承認プロセスへと持ち込まれた。

OSI 承認プロセス (図 8) では、OSI にライセンスをレビュー後、Web サイトにおけるオープンなレビューの場で OSD への適合性が審査され、その議論の中でライセンス内容や表現を調整し、最終的に議論の結果をふまえて OSI のボードミーティング (役員会) で OSI 承認の可否が決定される。

³⁶ IPA フォントライセンス v1.0 : <http://ossipedia.ipa.go.jp/ipafont/>

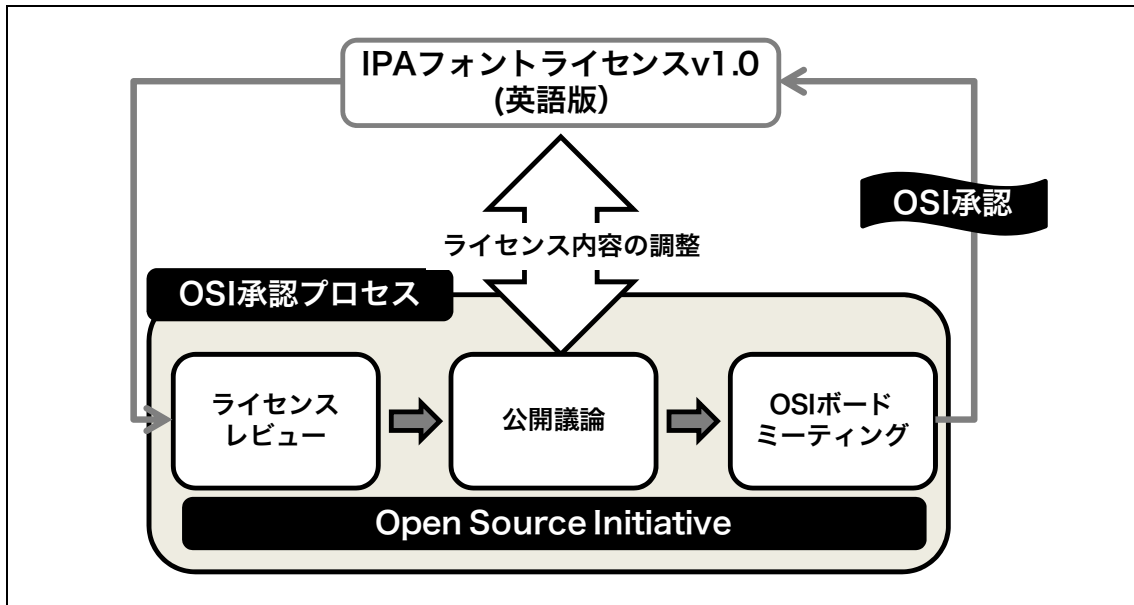


図 8 OSI 承認プロセス

「IPA フォントライセンス v1.0」は、2009 年 2 月 4 日から 3 月 31 日までのレビュー期間における議論と修正を経て 2009 年 4 月 1 日に OSD に準拠していることが認められ、オープンソースライセンスとして正式に OSI に承認された³⁷。フォントに特化したライセンスとしては、初めての OSI 承認ライセンスである。

レビュー期間に公開議論された主なポイントは以下の 2 点である。

(1) OSD2 項目「ソースコードが入手可能」への対応

「5. 2. 1 OSI 承認」で記述したとおり、フォントにおいてはソースコードという概念が明確ではない。従って IPA フォントにおいては、IPA フォント自体がソースコードでもあるという主張を行った。

これまで、OSS にはソースコードがあることが前提で議論されてきており、レビュー当初、OSI ボードメンバーによる理解を得るためにフォント制作についての説明が必要となったが、結果的には、IPA サイドの主張が認められた。ただし、フォント制作工程によっては、フォント制作ソフトウェアが独自に作成するネイティブファイルや中間ファイルが生成されることもあるため、「派生フォントには、それをさらに改変するために必要となる十分な情報を添付しなければならない。」ことを明

³⁷ OSI 承認ライセンスの一覧：<http://opensource.org/licenses/category>

記した。

(2) 派生フォントの頒布の制限

OSD を満たすためには派生フォントの許諾が必須である。しかし、OSD においても自由の概念を満たす趣旨において頒布の制限は認めている。オリジナルフォントが保証する文字への信頼性確保のため、どのようにして頒布制限と OSD を満たすかが最大のポイントとなった。「IPA フォントライセンス v1.0」は、2009 年 2 月 4 日のレビュー段階では、「派生フォントは、オリジナルのフォントからの変更部分を派生フォントとは分離し、(OSD 4 項目で認めている) 差分ファイルの形式により再配布するものとし、オリジナルのフォントを改変した状態で派生フォントを再配布してはならない。」として、作者のソースコードの完全性 (integrity) を確保する方式を規定していた。

この差分ファイル方式は、OSD2 項目「ソースコードが入手可能」の精神に沿ったオープンな C 言語のようなソースコードの存在が前提になり、バイナリについては、差分ファイル方式は認められない。従って、バイナリ要素も併せ持つフォントにおいては差分ファイル方式の許諾は困難であろう、という OSI ボードメンバーの意見が主流となった。このため、派生フォントの頒布の制限方法を検討し直し、「(2) 派生フォントの頒布とその制限」に示したように、「利用者が、その意志により派生フォントをオリジナルのフォントに戻せる方法を提供しなければならない。」、「派生フォントには、それをさらに改変するために必要となる十分な情報を添付しなければならない。」、「派生フォントにも、同じ IPA フォントライセンスを継承しなければならない。」、「派生フォントは、Web 等のだれもがアクセスできる方法により一般公開しなくてはならない。」ことを派生フォントの頒布制限とした。本改訂をレビューし、公開議論の下、OSI ボードメンバーに OSD 条件を全て満たしていると判断され、OSI ボードミーティングで承認された。

これにより、本研究で検討したフォントにおける DRE は、国際的慣習に整合性の有るオープンソースライセンスとして認められたことになる。

時を同じくして、SIL の OFL がフォントライセンスとして OSI 承認された。OFL は、2008 年 10 月に OSI 承認プロセスに入ったが、半年に及ぶ公開議論の結果、IPA ライセンスと同時に承認となった。OFL と IPA フォントライセンスの違いは、商用配布に対するスタンスの違いである。OFL は、フォント単独での商用配布を認め

ておらず、OSDの10項目のうちの「個人やグループに対する差別の禁止」に反することが承認のネックとなっていた。しかし、プログラムと同時に配布すれば商用も許諾するという記述により承認に至った。「IPA フォントライセンス v1.0」では、フォント単独での商用配布を許諾しており、利用分野に対していかなる差別も行っていない。

4. 6 おわりに

本研究の第1の貢献は、相互運用性に富んだ情報通信基盤の確立へ寄与する日本語パブリックフォントの定義付けを行ったことにある。第2の貢献は、オリジナルフォントが保証する文字への信頼性確保の要件を明確にし、日本語フォントと情報通信基盤との関係性を明らかにした点にある。第3の貢献は、文字への信頼性確保の要件と、OSIによるOSDのオープンソース定義とを両立させてDREの視点に立脚したライセンスとして日本で始めて具体的要件を構築しOSI承認を得たことにある。結果として日本語フォントをデジタル財としての自由に流通できる環境を整えることができた。

具体的には、フォントのDREとして、日本語パブリックフォントライセンスの確立をおこなった。

これまで、オープンなデジタル財のDREとしては、プログラムを対象としたGPLや、コンテンツを対象としたクリエイティブ・コモンズなどが確立しており、さまざまなデジタル財に用いられてきているが、日本語パブリックフォントに適したライセンスは存在していなかった。本研究では、日本語フォントを分析し、日本語パブリックフォントライセンスとしての要件を整理する中で、フォントにおいては、派生フォントの「フォント名」に対する使用許諾範囲が重要であることを明らかにした。また、情報通信基盤としての日本語フォントにおいてユーザーの混乱を招かないように、オリジナルフォントが保証する文字への信頼性を確保することが重要な要件であることを明らかにし、そのための方法を検討・提案した。

本研究で検討した日本語パブリックフォントライセンスである「IPA フォントライセンス v1.0」は、OSI承認されたことにより、国際的慣習に整合性の有るオープンソースライセンスとして認められた。従って、「IPA フォントライセンス v1.0」はIPAフォントや日本語フォントだけでなく国際的に利用可能なライセンスだと言

える。日本語パブリックフォントと同様にフォントへの信頼性を確保したいと考える他の言語のフォントにおいても充分適用可能なライセンスであると言える。

なお、本研究で OSI 承認された「IPA フォントライセンス v1.0」に関する筆者の位置付けとしては、IPA の研究員として「IPA フォントライセンス v1.0」の策定の主要メンバーとして活動するとともに、OSI 承認に向けた交渉の窓口を担当した。OSI 承認に向けた交渉は、「IPA フォントライセンス v1.0」が法的に矛盾のないものとするために IPA が公募して採用した弁護士を交渉の前面に立てたが、筆者は公開議論の過程で OSI ボードメンバーからの意見に対する対応を弁護士と調整し、弁護士が公開議論で交渉を行う上での材料を提示する役割を果たした。この公開議論の全記録は、OSI の Web サイト³⁸に公開されている。

「IPA フォントライセンス v1.0」の確立により、IPA フォントはより幅広い OSS での活用が期待できる。情報通信基盤となるべきパブリックフォントとして、日本語フォント特有の品質維持、間違っただ字形の氾濫阻止が行え、フォントの改変と再頒布が可能なパブリックフォント概念の創出と日本語フォントの健全な流通に貢献するものである。

今後は、日本語パブリックフォントライセンスが及ぼした影響を、IPA フォントを通して、長期スパンで観察し、評価する。観察内容としては、IPA フォントの利用者数の推移、利用シーン・利用媒体の展開、派生フォントの状況、不正使用の有無などが考えられる。

また、日本語パブリックフォントのメンテナンス活動をフォローし、より良い日本語環境の基盤となる情報発信、情報共有のあり方を検討提案し実践することで、日本のソフトウェア産業、日本語を使う全ユーザーへのフィードバックを行う。

³⁸ OSI の Web サイトの中の、Mailing Lists のアーカイブにこれまでの OSI 承認における公開議論の記録が公開されている。「IPA フォントライセンス v1.0」については、Public Lists の License Review の中の以下のアドレスで確認できる。
<http://www.crynwr.com/cgi-bin/ezmlm-cgi?17:ddn:0:200901#b>

第五章 グリフデータベースの構築

5.1 はじめに

第三章、第四章では情報通信基盤としての日本語パブリックフォントについて、運用面、ライセンス面で提案を行ってきた。本章では、日本語パブリックフォントだけでは解決できない異体字や外字と言った文字問題へのソリューションについて検討を行い、システム構築を行った。

5.2 異体字・外字への取り組みの必要性

日本語パブリックフォントは、相互運用性の高いフォントを提供することを重視して、フォントがオープンであることすなわち文字コードとしてもフォントフォーマットとしてもデファクトを採用することで、品質を保持するという考えのもと設計してきた。

しかし、日本語の情報通信基盤には、各官公庁、地方自治体を含めた行政システムにおけるいわゆる外字問題が立ちはだかってきた。外字問題とひとくくりにされているが、大きく二つに分けられる。

一つは、JIS X 0208 や ISO/IEC 10646 および Unicode などの情報交換用符号化文字集合では包摂規準や Unification Rules のために社会慣習上使い分けられている字形（＝異体字）が区別できないという問題である。もう一つは、そもそも符号化文字集合上に符号位置を持たない文字（＝外字）をどう取り扱うかという問題である。

図 9 に示すように、異体字は情報交換用符号化文字集合に収納されている文字の包摂関係にある文字であるのに対し、外字は情報交換用符号化文字集合には属さない、つまり情報交換用符号化文字としては認められないクローズドな環境で使用する文字である、ということが出来る。

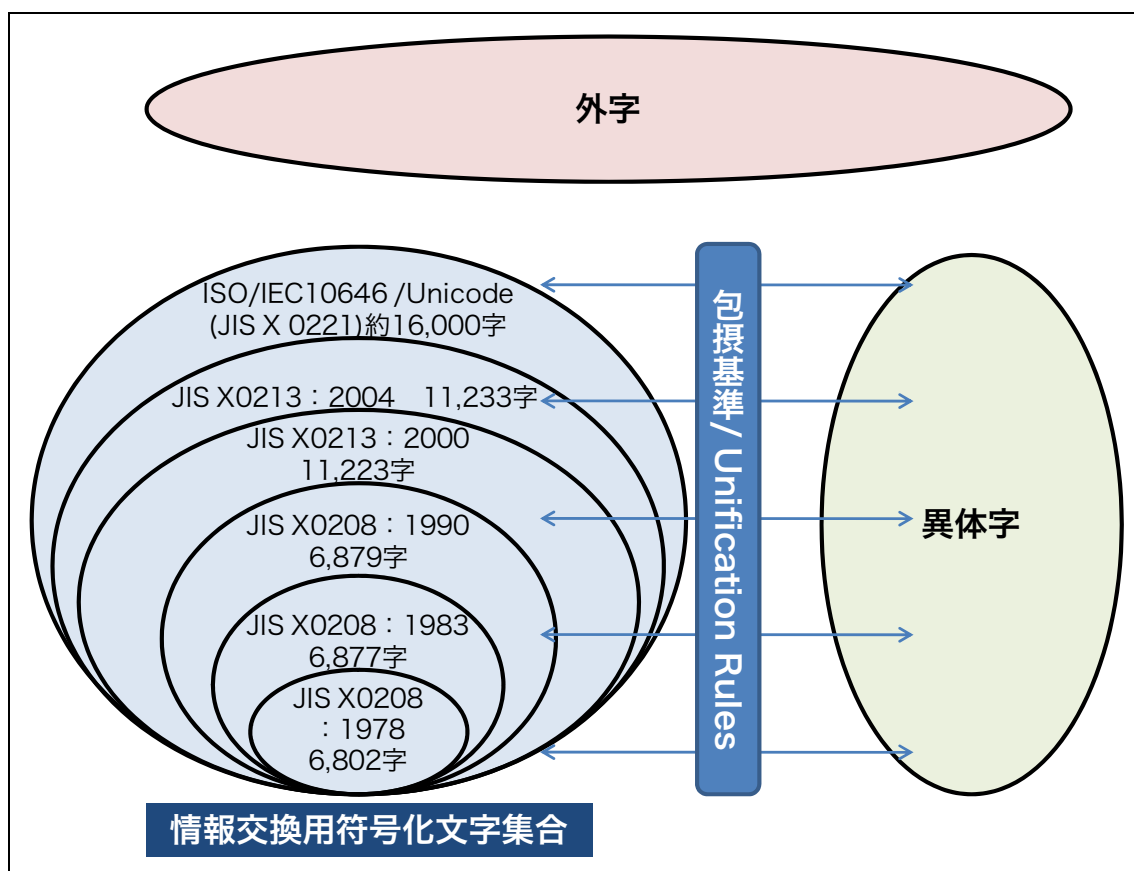


図 9 異体字、外字の関係図

いわゆる外字問題には、決定打となるような標準的ソリューションがなかったため、私用領域³⁹を用いたり、フォント名が異なるフォントファイルの恣意的な位置に字形も意味も全く異なる文字を割り付けたり、符号化せずに画像として表示したりといった、さまざまな方法が林立してきた。以前の紙にさえ印刷できたらよい時代ではこれらの手法でしのいでこられたが、ネットワーク環境における情報資源としての活用の必要性が高まるにつれ、電子化文書としてのデータの互換性、検索性、相互運用性を著しく妨げることとなってきた。それぞれの独自システム開発のために投下されたコストも膨大な額に上っている。

情報交換性や相互運用性といった観点からは、文字の使用を情報交換用符号化文字集合の範囲内に止めることが望ましいが、地名や人名の表記に関しては、居住地への愛着心や父祖への尊敬心といった個人のアイデンティティに関わることでもあ

³⁹ Shift JIS では、xF040~F9FF を外字領域と定義している。Unicode では、0xE000-F8FF を Private Use Area (私用領域) と定義している。

り、字形の使い分けへの要求を拒否することも困難である。

したがって、異体字、外字を我が国の情報通信基盤としてどのように扱うべきかという提案が求められる。

5. 3 異体字・外字ソリューション

5. 3. 1 異体字

これまで、情報交換用に使用するフォントのフォーマットとしては、JIS X 0208 や ISO/IEC 10646 および Unicode などの情報交換用符号化文字集合に符号位置を持つ文字を、1つの符号位置に対し1つの字形しか実装することができなかった。日本語フォントの多くは、規格表に印刷された例示字形と包摂規準と各フォントのデザインポリシーに従って、1つの符号位置に割り当てる字形を作成している。つまり、あるフォントにおける符号位置 a に割り当てられる字形は「Ga」1つであり、「Ga」と包摂関係にある字形「Ga'」や「Ga”」は、同じフォントでは表示することができない。ここで、「Ga'」や「Ga”」は「Ga」の異体字である。あるいは、別のフォントにおいて「Ga'」を収納する字形として採用した場合は、「Ga'」の異体字である「Ga」や「Ga”」は表示することができない。

異体字に関しては、ISO/IEC JTC1/SC2⁴⁰および Unicode Consortium による標準化活動の成果によって、国際標準に準拠したアーキテクチャでの技術的な解決として IVS、IVD が示された事で、異体字のフォントへの実装の道が開けた。

IVS は異体字等を指定するための機能であり、異体字セレクタ (Ideographic Variation Selector Character : IVSC) を使って、Base Character (Unicode に登録されている文字) に対する異体字を指定することができる (図 10)。異体字セレクタは U+E0100 から U+E01EF までを割り当てられており、技術的には Base Character と異体字セレクタの組み合わせで 240 の異体字を表現することが可能となる。

IVD は、異体字セレクタを実際に漢字で運用するための漢字字形データベースの規格である。IVS を利用して誰もが自由に異体字を追加したフォントを作成したのでは、情報交換上混乱をきたすのはあきらかである。そこで Unicode Consortium

⁴⁰ ISO/IEC JTC1/SC2 は、国際標準化を行う ISO と IEC の合同委員会 (ISO/IEC JTC1) において、符号化文字集合に関する標準化を担当するサブ委員会である。

では、ある特定の団体が IVS の集合に名前をつけた Ideographic Variation Collection (IVC) と IVS を登録する枠組みとして IVD を規格化し、審査・承認のプロセスをとるようにしている (図 10)。現在のところ、2007 年 12 月 14 日に承認された Adobe-Japan1⁴¹が IVD に唯一登録されている。この Adobe-Japan1 の IVD には 14,647 グリフが収納されている。

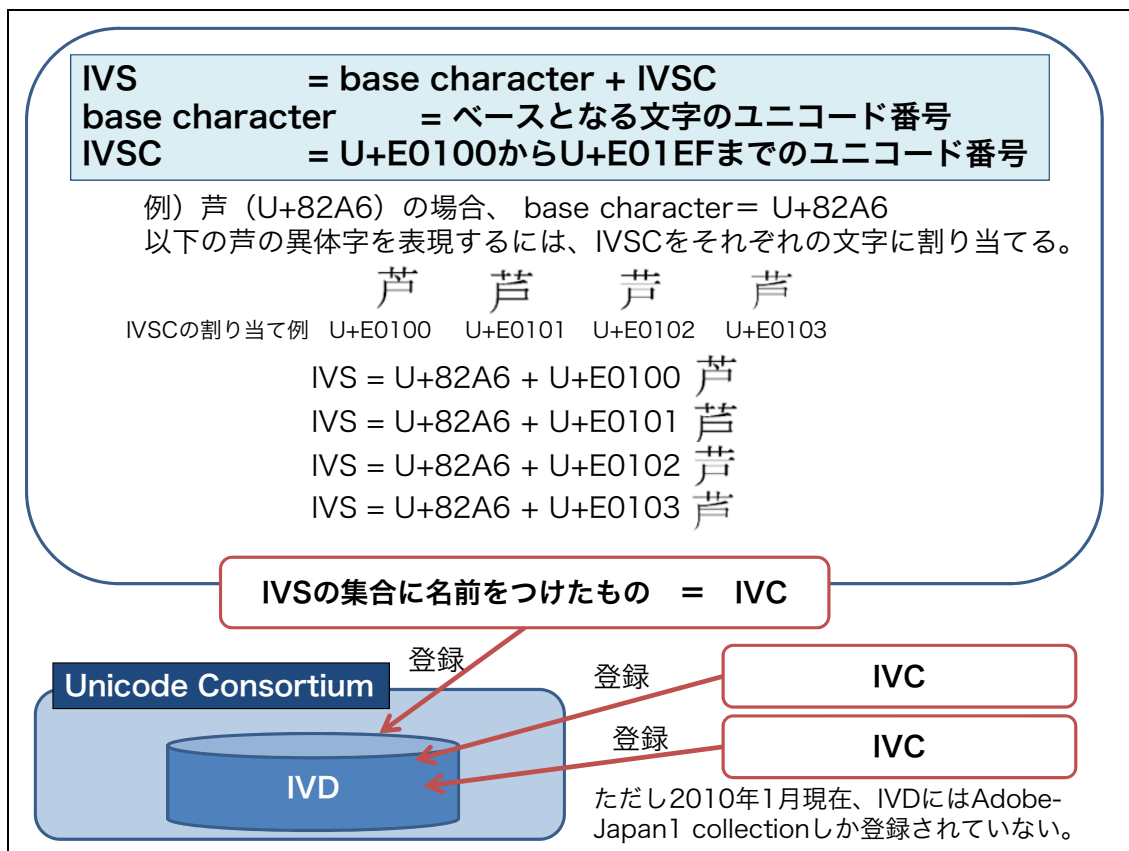


図 10 IVS,IVSC,IVD の関係

図 11、図 12 は、IVD 登録されている Adobe-Japan1 の一例である。「辻」は、一点しんにようと二点しんにようで包摂関係にある。従って、Unicode 上のコードポイントは U+8FBB だけであり、一点しんにようと二点しんにようを同一フォントで表示することはできなかった。IVS を用いることで、

U+8FBB U+E0100 =一点しんにようの辻

⁴¹ Unicode Home Page 「Combined registration of the Adobe-Japan1 collection and of sequences in that collection」：
<http://www.unicode.org/ivd/data/2007-12-14/>

U+8FBB U+E0101 = 二点しんによりの辻
を切り替える事が可能となる。

「龍」についても同様で、微妙な字形差を区別したい場合には、U+E0100、U+E0101、U+E0102 の 3 種類の IVS により字形を切り替える事ができる。



図 11 Adobe-Japan1 IVD における辻の例

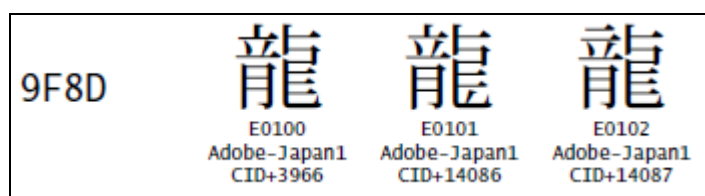


図 12 Adobe-Japan1 IVD における龍の例

実際に IVS を利用して異体字を出力するためには、文字処理環境として、入力 (InputMethod)、処理 (各種アプリケーション)、出力 (フォント、レンダリング) のすべてが対応しなければ成立しない。

そのためには、フォント側の環境として、IVS を利用した異体字追加フォントを作成しやすい環境作りが求められる。フォントへの異体字追加には、異体字をゼロから作るよりもすでに登録されている字形やそのパーツ (偏や旁など) を利用した改変を行う方が、効率的である。また、同一の字形がすでに作成されていないかを確認する必要もある。日本語フォントは、多くの文字を収納しているため、所望の字形を探し出すことが難しい場合がある。特に読みのわからない文字については、検索する手掛かりが無い場合がある。ここで課題として、字形の検索という問題が浮かび上がる。

一方、情報の交換性、相互運用性観点からは、異体字として表示できる符号化ソリューションがベストであるが、入力 (InputMethod)、処理 (各種アプリケーシ

ョン)、出力 (フォント、レンダリング) のすべての環境が IVS に対する技術的対応を必要とする。多くのフォントに同様の実装がされていなければならない、多数の異体字を搭載した巨大なフォントを作成するよりも、異体字を個別の文字画像として扱うことも 1 つの手段といえる。過渡的な状況や、異体字にも存在しない外字の文字表示要望に対しては、符号化しない外字ソリューションの提供が求められる。

5. 3. 2 外字

外字は、情報交換用符号化文字集合上に符号位置を持たない文字である。つまり、情報交換を行う上でオープンな環境において合意がとれていない文字であるといえる。

これまで、ある限定的な使用範囲で外字を使うにあたっては、空いている符号位置に外字を割り当てたフォントを使用することを行ってきたが、オープンな環境では情報交換や相互運用性といった観点から、外字をフォントに搭載するのではなくあくまで文字画像として扱うことが望ましい。これは、「4. 4. 2 フォント向けライセンスの考察」で示したようにフォントが OS やアプリケーションとコンテンツの間で、フォント名により連携をとっていることと関係する。

図 13 に外字をフォントに搭載した場合と、画像として扱った場合の例を示す。図 13 上部に外字をフォントに搭載した場合を示す。外字搭載フォント Fa を所有している A さんが外字を含むドキュメント Da を作成しオープンな環境に対して公開した場合、Fa を所有しない B さんが Da を閲覧する場合には、B さんの PC 環境に存在するフォント Fb を代替使用することになる。この際、符号化文字集合 (例えば JIS X 0213) 内の文字は問題なく表示できるが、外字については Fb には搭載されていないため表示することができない。この現象を情報交換性の欠落といえることができる。一方、図 13 下部に示すように、外字を画像として扱った場合は、A さんが作成したドキュメント Dc には外字画像が張り込まれており、それを受け取った B さんも外字画像が含まれたドキュメント Dc を閲覧するため外字部分の欠落は生じない。

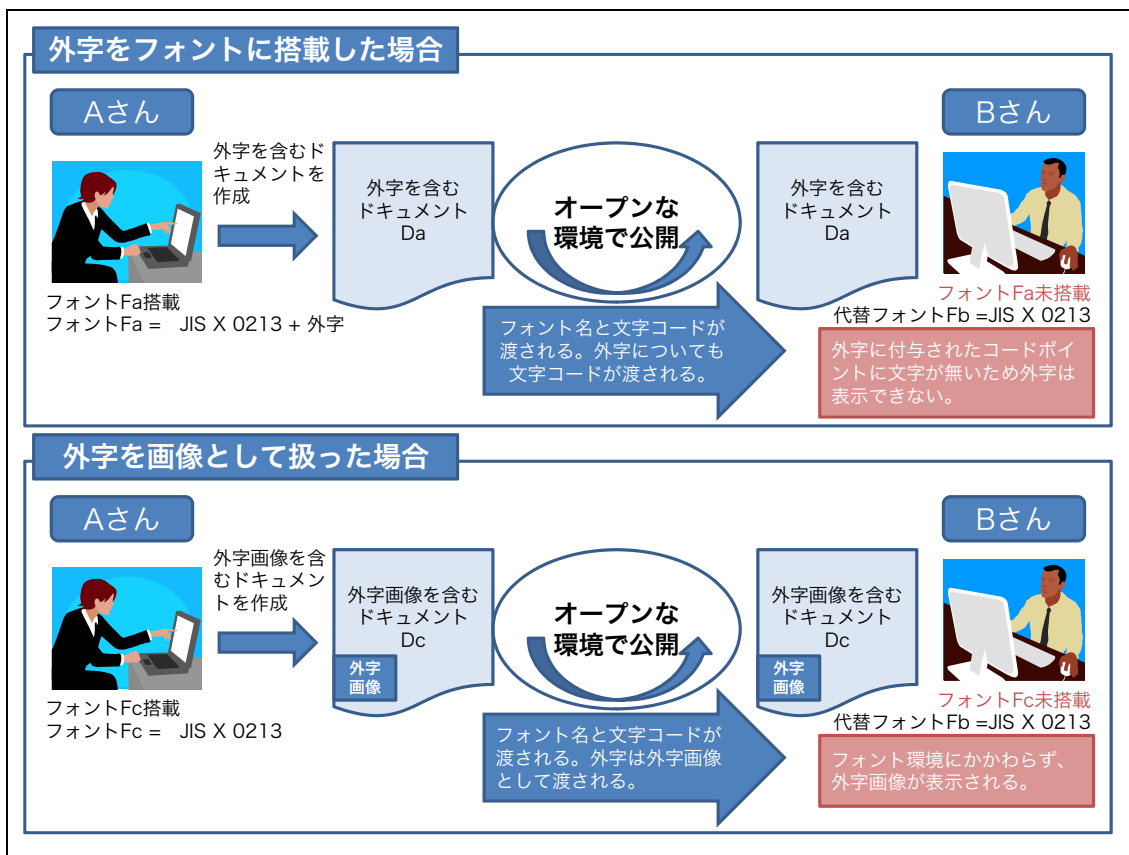


図 13 オープンな環境での外字の扱い方

外字を文字画像として扱うにあたってオープンであることが求められる。今後のネットワーク環境で自由度が高く、文字列の中に違和感なく埋めこむことの出来る汎用的なフォーマットであることが要件となる。

また、文字画像とした外字を、情報通信基盤として活用できる環境作りとしては、オープンなネットワーク環境に文字画像を公開し、そこから外字を引き出す手段を設ける必要がある。異体字と共通する課題として、字形の検索を行うためのシステムが求められる。

5. 4 グリフデータベースの提案

本研究では、日本語の情報通信基盤として、各官公庁、地方自治体を含めた行政システムでのテキスト情報資源を情報交換性および相互運用性の高いものとするための提案を行う。

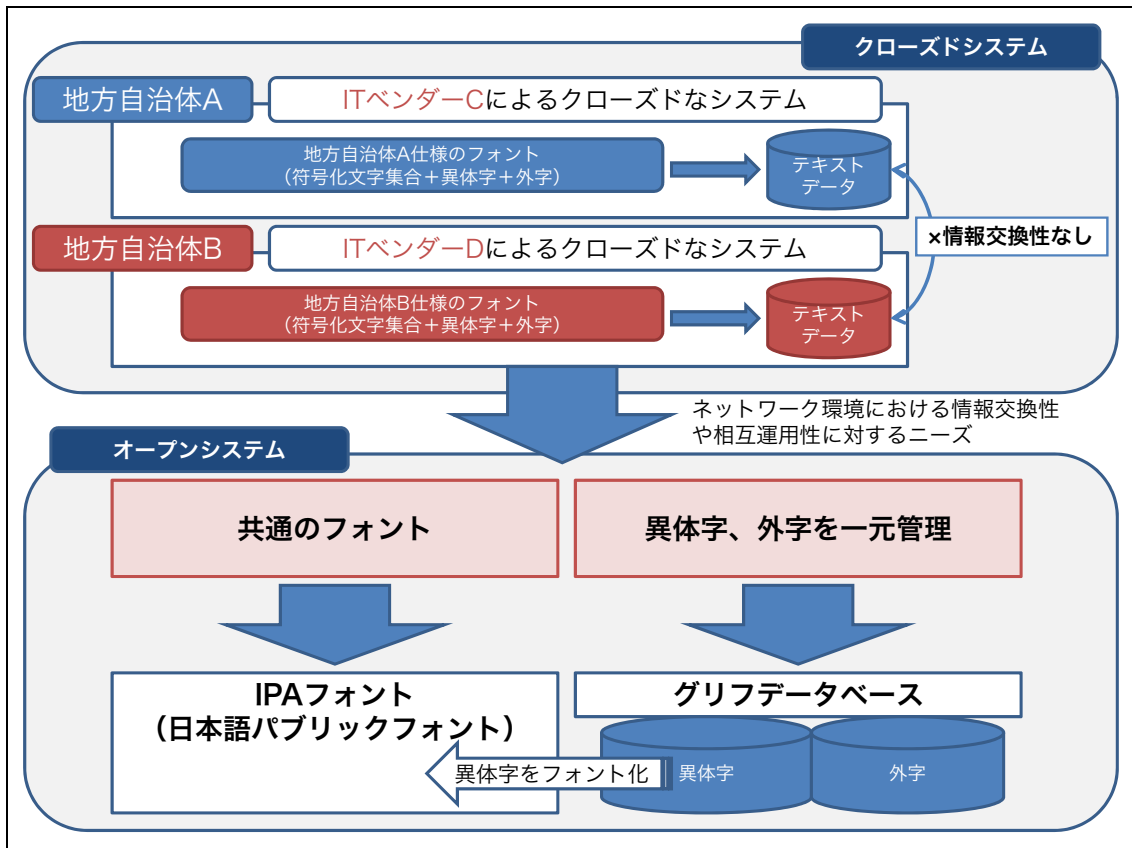


図 14 行政システムへの提案

図 14 に示すように、メインフレームを主流とするクローズドな環境で行政システムが運用されていた時代は、日本語フォントもハードウェアの付随品としてハードウェアベンダにより提供されるものであり、外部との互換を考慮せずクローズドな環境で表示・印刷できるものであれば良かった。従って、異体字、外字についてもクローズドな環境でのニーズに応え、適当に空いている符号位置に字形を割り振る対応をしており、異なる自治体間でのテキストデータには情報交換性はない状態であった。

しかし、インターネットによるネットワーク環境の充実は、クローズドシステムから電子政府・電子自治体というオープンなシステムへの展開を余儀なくされた。当然、その中で使用されるテキストデータは共通の符号化文字集合により運用されなければ情報交換性や相互運用性を担保されない。そのテキストデータの情報交換性や相互運用性の基盤の1つとして、符号化文字集合内の文字を収納したフォントが必要とされ、それが本研究で取り上げた日本語パブリックフォントである IPA フ

ントがある。そして、異体字、外字についても、オープンなシステムの中でどのように扱うかが、議論となる。

これまで、情報規格調査会(2002)により本論文で示す異体字、外字を「異形字⁴²⁾」と捉え、異形字アーキテクチャについて提案された経緯がある。情報規格調査会(2002)は、情報処理学会情報規格調査会に設置された文字コード標準体系専門委員会によってまとめられた報告書である。異形字アーキテクチャの提案では、符号枝番異形字拡張とフォント枝番異形字拡張が挙げられており、符号枝番異形字拡張はまさに IVS により Unicode 標準化されたものである。フォント枝番異形字拡張については、「今昔文字鏡⁴³⁾」という商用のフォントと漢字検索システムのパッケージソフトウェアでの実装があるが、デファクトスタンダードとしての利用は見受けられない。情報規格調査会(2002)では、異形字は文字の微妙な違いを扱うだけに、関係者の合意を得るのが難しい面があるため、今後の標準化の進め方に注意を要する課題であるとしている。また、文字データベースを作成して公開維持管理することを提案している。ISO/IEC 10646 の開発において収容文字数が 7 万字に近くなると、文字の同定やユニフィケーション作業が膨大な手間になり、品質の確保が難しくなるという経験から、文字コードの対象が専門的、学問的になるに従って分散開発・分散保守の必要性とされる。文字データベースには、部首・画数・読み・各種規格符号位置などのほか、典拠・用例、あるいは採録不採録の理由などのデータをできるだけ収集することが提案されているが、文字データベースの公開には至っていない。

他にも異体字、外字を含む多数の文字を扱うシステムとしては、上地(2008)の漢字字形情報管理システム「GlyphWiki⁴⁴⁾」がある。GlyphWiki は、Web データベースの形態をとり、インターネット上でグリフの検索、登録や作成を行うことができるとともに、グリフを集めてフォント化する機能を備えている。現在、登録・利用は無料であり、不特定多数のユーザーがシステム利用可能である。上地(2008)によ

⁴²⁾“字体”“字形”が異なるが、意味や読みなどが同じ漢字で、文中でそれらを入れ替えても善意の第三者にとっては同じ意味の文になる漢字。従来の符号化文字集合の枠組みの中では、包摂若しくは unify されることもある(情報規格調査会、2002)。

⁴³⁾ 今昔文字鏡：<http://www.mojikyo.org/html/index.html>

⁴⁴⁾ GlyphWiki (グリフウィキ) は、<http://www.fonts.jp/glyphwiki/>で公開されている。

ると GlyphWiki は、日本漢文学研究者が、ネットワーク時代において他者が提供する外字フォントを使うという受動的な立場にあるのは「インターネットの普及に代表されるコンピュータ同士の情報交換に対する外字利用の弊害」と「漢字の外字フォント作成やフォントの配布が面倒である」という要因であるにとらえ、それを解消するために研究開発されたものである。GlyphWiki には、すでに 10 万字を超えるグリフが収納されているが、だれもが登録できることからグリフのクオリティやまったく新規の漢字を作字して登録してしまうなど統制がとれていない部分もあり、学術用途の多様なグリフを表現するには有意義であるが、各官公庁、地方自治体を含めた行政システムの基盤としては、冗長なシステムであるといえる。

既存研究において情報規格調査会(2002)、上地(2008)ともに、異体字、外字を含む多文字を扱うためには、公開での文字データベースの利用が不可欠とされている。

また、政府レベルでの取り組みの 1 つに、法務省の戸籍統一文字がある。戸籍統一文字は、戸籍に関する届出、または戸籍に関する証明書の交付申請をインターネットで行うために整理した文字である（法務省、2004）。日立(2007)によると、戸籍に使用できる文字として法務省が制定した 56,040 文字とされ、異なるコンピュータ間での正確な文字伝達・交換を実現するために、それぞれの文字には戸籍統一文字番号、字形情報、読み、画数、部首などの属性情報が付与されている。また、戸籍統一文字は、平成 16 年 4 月 1 日付法務省民一第 928 号民事局長通達の別冊（CD-ROM）として法務省（法務局）から市区町村へ配布されている。WEB でも「戸籍統一文字情報⁴⁵」として一般に公開している。しかしながら、戸籍統一文字は戸籍統一文字番号により独自の文字集合を制定したに過ぎず、オープンに利用可能なフォントの提供があるわけではなく、WEB で公開されている文字は PNG 形式の画像データであるため、他の行政システムで汎用的に利用可能な状態にはない。

そこで本研究では、行政システムに適したオープンなシステムでの異体字、外字の一元管理を GlyphWiki と同様に、Web データベースでの管理にとらえ、インターネット上でグリフの検索、取得を行うグリフデータベースを提案する。「5.3 異体字・外字ソリューション」では、異体字を IVS によりフォント化するアーキテクチャを示したが、現在は IVS を利用して異体字を出力する環境が十分に整っていない

⁴⁵戸籍統一文字情報の Web ページ：
<http://kosekimoji.moj.go.jp/kosekimojidb/mjko/PeopleTop/>

いという過渡的な状況にあるため、当面は異体字を外字と同様に文字画像として配信することを前提とし、行政システムにおいて既存のグリフを検索、取得し活用可能なシステムとする。また、登録は誰でも行うのではなく、管理者が行うことで行政システムでの品質保持を保つものとするが、管理者が新規グリフを作成する際に必要となる既存グリフのパーツを検索しやすいシステムを目指す。

提案システムは、自治体の事務処理窓口で文字を扱う担当者と新規グリフを作成する管理者を対象とする。自治体の事務処理窓口での検索したグリフの利用は Web プラットフォームを中心とすることから、グリフのデータ形式は Web と親和性の高いものが要求される。

以下にグリフデータベースの要件を列記する。

- ① グリフを容易に検索できること。
- ② 異体字作成に必要なパーツを容易に検索できること。
- ③ そのグリフを文字画像としてオフィスドキュメントや Web 画面上で使用できること。
- ④ そのグリフをフォントとして登録するためのデータとして使用できること。

このうち、①②については、データベースに収納するグリフに付与するメタデータの定義がカギとなる。③④については、オフィスドキュメント、Web 画面上およびフォント作成ツールで使用可能なグリフ用の画像フォーマットの研究が必要となる。

グリフデータベースとは、グリフ字体とそのグリフに付与するメタデータをデータベース管理し、データベースへのグリフの登録およびグリフの検索とダウンロードを行うためのシステムである（図 15）。

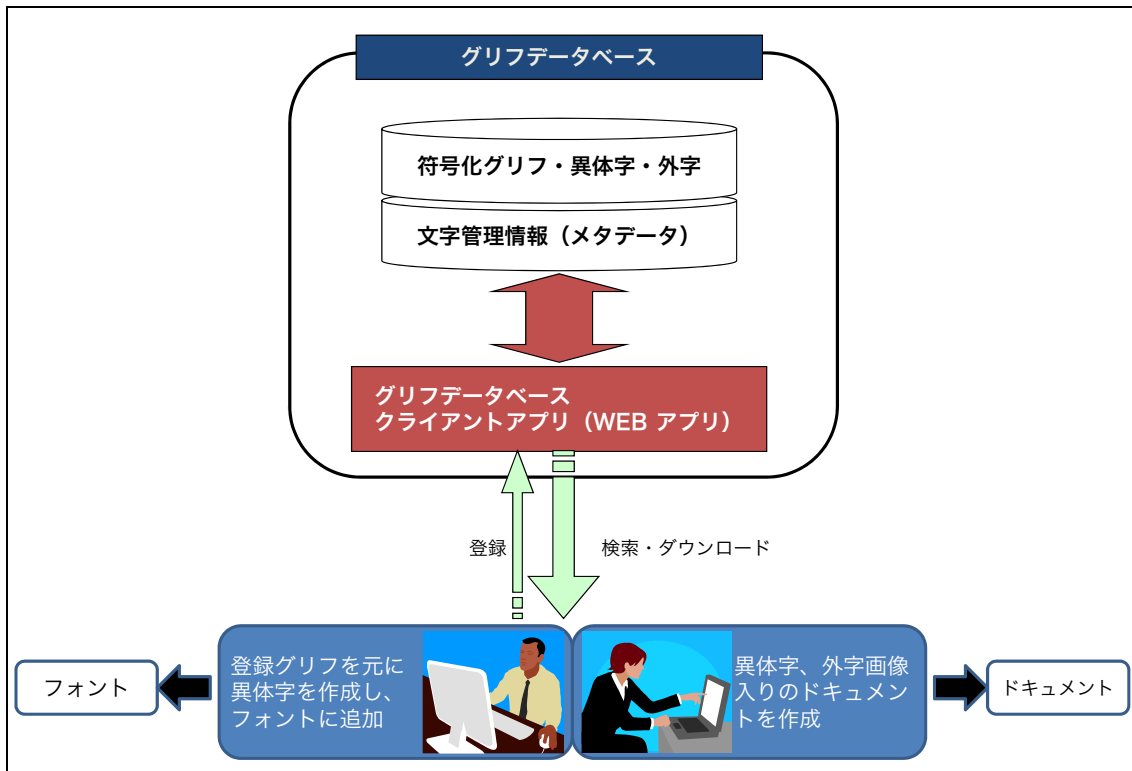


図 15 グリフデータベースの概要

5. 4. 1 メタデータの検討

グリフをデータベースに収納するにあたり、各グリフを管理し検索するための情報が必要となる。グリフデータベースが格納対象とするグリフとは、すでにフォントに収納されている符号化文字集合上に符号位置を持つ文字と、その異体字、および符号化位置を持たない外字である。検索者は、必ずしもそのグリフの読みや文字コードがわかっているとは限らないため、出来るだけ多くの文字管理情報（メタデータ）を付与することで、所望のグリフの検索性を高めることができると考える。

先行研究や既存のシステムとして、情報規格調査会(2002)、上地(2008)、法務省(2004)において提案されている文字情報を参考とし、グリフに対するメタデータとして、(1) 文字コード情報、(2) 文字情報、(3) 登録情報、(4) 利用情報、の4種類についてそれぞれ検討した。

(1) 文字コード情報

収納対象とするグリフに付与可能な文字コードならびに文字番号としては、

Unicodeをはじめとする国内外の符号化文字集合における文字コードと、符号化文字集合を策定時に参考とされる康熙字典⁴⁶の文字番号と、大漢和辞典⁴⁷の大漢和番号を付与する（表 10）。

表 10 文字コード情報

メタデータ	説明
グリフID	DBが生成するユニークなID（完全ユニーク）
Unicode	16進の5桁～6桁（例：U+82A6）
IVS	異体字用（例：82A6 E0101）
UTF-8	Unicodeから変換（例：E88AA6）
JIS面区点	JIS X 0213の符号位置（面-区-点）（例：1-16-18）
JIS	JISコード（例：3032）
EUC	Extended UNIX Code（例：B0B2）
SJIS	Shift-JISコード（例：88B0）
Adobe-Japan 1-6	Adobe-Japan-1-6のCID番号（例：7961）
大漢和番号	大漢和辞典の文字番号（例1：17920、例2：17920'、例3:ho310）
康熙字典番号	康熙字典（同文書局版）のページを基本にした文字番号（例：0084.141）

表 10 の内、グリフ ID はデータベース登録時にユニークに付与される ID である。また、IVS は登録グリフが異体字の場合のみ必要なデータである。グリフ ID と IVS 以外は、Unicode または Adobe-Japan 1-6 との対応テーブルにより、一意に決定できるコードおよび番号である。従って、グリフ登録時に文字コード情報をすべて入力する必要はなく、グリフデータベースが対応テーブルを保持することで、Unicode

⁴⁶中国、清の康熙帝が編纂させた字書で 1716 年に完成した。いくつかの版があり、複数の部首に再掲載されている文字もあるので、全体の収録字数は 40,000 字～50,000 字の間と考えられる。歴代の中国の字書の総括という評価を与えられ、漢字の字体の上で今も規範をなす典拠とされることが多い。また例字は明朝体であり康熙字典体と呼ばれる。康熙字典体は幕末に清から輸入した明朝体活字のデザインの元となったといわれるため、日本の明朝体活字にも大きな影響を及ぼしている。これまでに、Unicode の規格化や国語審議会の表外漢字字体表作成時に康熙字典体が規範として参照された経緯がある。

⁴⁷ 親文字 5 万余字、熟語 53 万余語を収録した世界最大の漢和辞典。諸橋轍次著による「大漢和辞典」であり、諸橋大漢和と呼ばれることもある。

または Adobe-Japan 1-6 の入力により自動的にその他のコードおよび番号を発生することとする。

(2) 文字情報

文字情報とは、そのグリフの文字としての属性を表すメタデータであり、表 11 のデータを対象とする。

表 11 文字情報

メタデータ	説明
名前	Unicodeコードチャート上の文字の名前
分類見出し	漢字か非漢字かの分類見出し (JISX0213による分類)
書体識別	明朝かゴシックか共通
縦書き置換	縦書き用のグリフなのかどうか
総画数	文字の画数 (Stroke Count) (漢字のみ)
部首	部首
音読み	読み方 (音読み)
訓読み	読み方 (訓読み)
異体字	そのグリフにUnicode上の別コードに符号化されている異体字がある場合、ユニコードで記述
IDS	IDS : Ideographic Discliption Sequence

この内、書体識別と縦書き置換についてはグリフ毎に設定が必要であるが、それ以外は、「(1) 文字コード情報」の中のメタデータである Unicode または Adobe-Japan 1-6 との変換テーブルにより、一意に決定できる。「(1) 文字コード情報」と同様に対応テーブルをグリフデータベースに保持することで自動発生することとする。

なお、IDS (Ideographic Description Sequence)⁴⁸とは、文字校正記述文字 (Ideographic Description Characters : IDC、U+2FF0～U+2FFB) を用いて、漢字の構成を表す文字列である。IDC は、Unicode Inc.(2009:296)により図 16 のように定義されている。

⁴⁸ <http://www.unicode.org/versions/Unicode5.0.0/ch12.pdf#G15784>

Ideographic description characters		
<i>These are visibly displayed graphic characters, not invisible composition controls.</i>		
2FF0		IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO RIGHT
2FF1		IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO BELOW
2FF2		IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO MIDDLE AND RIGHT
2FF3		IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO MIDDLE AND BELOW
2FF4		IDEOGRAPHIC DESCRIPTION CHARACTER FULL SURROUND
2FF5		IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM ABOVE
2FF6		IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM BELOW
2FF7		IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LEFT
2FF8		IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER LEFT
2FF9		IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER RIGHT
2FFA		IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LOWER LEFT
2FFB		IDEOGRAPHIC DESCRIPTION CHARACTER OVERLAID

図 16 Ideographic Description Characters

文字情報の内、部首と IDS については、近似したグリフを検索する際の手掛かりともなるため、所望のグリフ検索の手掛かりとなると同時にグリフデータベースの要件として列記したうちの「②異体字作成に必要なパーツを容易に検索できること。」に対応するメタデータにもなる。

(3) 登録情報

登録情報とは、グリフを登録した際に付与されるメタデータであり、表 12 のデータを対象とする。

表 12 登録情報

メタデータ	説明
作成者	グリフ登録をしたユーザ
登録日時	作字情報の登録日時
最終更新日時	最終更新日時：初期値は、登録日時＝最終更新日時となる
読み方	自由入力する読み方置換情報
参照文字	グリフを作成したときに参照したグリフがある場合にグリフIDを記録、複数可
備考	典拠情報、使用事例等の備考欄

作成者については、システムの運用の問題となるが、事前に登録権限を持つユーザーには ID を付与し、そのユーザーID をメタデータとして保持する。登録日時、最終更新日時はグリフデータベースへのアクセス日時を自動発生する。読み方とは、「(1) 文字コード情報」では登録されない特殊な読み方や外字の読みを自由入力するためのメタデータである。参照文字とは、新規にグリフを作成した場合に、参照したグリフがある場合に登録するグリフ ID である。備考とは、グリフ登録時に「(1) 文字コード情報」、「(2) 文字情報」、「(3) 登録情報」の各メタデータ以外のグリフに関する情報を自由に記述するためのメタデータである。

(4) 利用情報

利用情報とは、事後メタデータと言うべき情報であり、グリフのダウンロード時に作成されるメタデータであり、表 13 のデータを対象とする。

表 13 利用情報

メタデータ	説明
利用者	ダウンロードしたユーザ名
使用内容	グリフを使用する用途：人名、地名、その他
使用メディア	グリフを使用するメディア：書籍、Web、その他
ダウンロード日時	グリフをダウンロードした日時
コメント	利用ユーザからのコメント
フォーマット	ダウンロードするグリフフォーマット：編集用SVG/SVG/EPS
利用回数	ダウンロード回数

この利用情報は、各グリフのニーズや利用者の使用目的を蓄積することで、社会的な文字に対するニーズの参考とするものである。

5. 4. 2 グリフフォーマットの検討

グリフデータベースに収納するグリフのフォーマットを検討する。

グリフデータベースに収納するグリフは、文字画像としてオフィスドキュメント、Web 画面上およびフォント作成ツールで使用する。

グリフ画像は、いずれの利用用途においても、表示サイズによって品質を損なわないアウトラインデータであることが望ましい。フォント制作の現場で良く使用されているフォーマットは EPS (Encapsulated Post Script) である。従って、グリフフォーマットの 1 つとして EPS を収納することとする。

一方、オフィスドキュメントや Web 上で、文字列の中に文字画像を混在して表示する技術として、SING (Smart INdependent Glyphlets) と SVG (Scalable Vector Graphics) を検討した。

SING⁴⁹とは、Adobe Systems 社が、文字コード規格と OpenType フォーマットだけでは解決できない文字として外字 (グリフ) をコンピュータ上で容易に再現するために考えたアーキテクチャである。「グリフレット」という 1 文字分の字形情報と付属情報 (親フォント、字幅、読み、言語など) 持つファイルを SING 対応アプリケーションに埋め込むことで、各工程でグリフレットの存在を気にすることなく、入力や編集、出力などに一貫性を持った作業が可能になる。しかしながら、2009 年現在、SING 対応アプリケーションは Adobe Systems 社製の DTP アプリケーションのみであるため、さまざまなオフィスドキュメントや Web 環境での使用には当面对応できないと判断し、SING フォーマットは採用しないこととする。

SVG⁵⁰とは、W3C (World Wide Web Consortium) で標準化されている XML によって記述されたベクターグラフィック言語である。W3C は、World Wide Web で使用される各種技術の標準化を推進する為に設立された非営利団体であり、SVG 以外にも HTML、XML 等の標準策定を行っている。2001 年に SVG 1.0 が W3C 勧告

⁴⁹ http://www.adobe.com/devnet/font/pdfs/5148.SING_Tutorial.pdf

⁵⁰ <http://www.w3.org/Graphics/SVG/>

となって以来、数回の更新がなされ、2008年12月22日にW3C勧告されたSVG Tiny 1.2⁵¹が最新バージョンである。SVG Tiny 1.2には17章目に「Fonts」が定義されており、Webブラウザ上で単一の文字（グリフ）を表示するための記述方法が明記されている。オープンな標準であるSVG Tiny 1.2に準拠することは、SVG対応のエディタやブラウザが増える可能性を期待できる。グリフデータベースに収納するグリフファイルはWebブラウザでの使用が最も優先されることから、SVG Tiny 1.2を採用することとした。

そこで、オープンなフォントであるIPAフォントをサンプルとして、IPA明朝フォントの「の」のグリフをSVGファイル化し、ブラウザ上で表示する実験を試みた。

まず、「の」のSVGファイルを作成した。ここでは「の」にid="GID13135"を付与している。

[GID13135.svg]

```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink"
    version="1.2" baseProfile="tiny" viewBox="0 0 1000 1000">
    <!--glyph metrics info-->
    <defs>
        <font horiz-origin-x="0" horiz-adv-x="1000" id="GID13135">
            <font-face font-family="IPAMincho"
units-per-em="1000" ascent="880" descent="120"/>
            <!--glyph points cordinate-->
            <glyph unicode="GID13135" glyph-name
="GID13135" vert-origin-y="786" vert-adv-y="870"
                d="M505 683
                Q512 642 512 585
                Q512 417 393 225
```

⁵¹ <http://www.w3.org/TR/SVGTiny12/>

Q338 137 266 88
Q236 70 214 70
Q178 70 146 107
Q91 169 91 279
Q91 450 218 582
Q348 716 528 716
Q666 716 762 649
Q901 553 901 378
Q901 193 767 98
Q668 27 488 16
L482 41
Q830 88 830 376
Q830 523 734 608
Q648 684 526 684
Q517 684 505 683

M263 137
Q267 140 273 146
Q346 213 400 353
Q449 481 449 621
Q449 651 446 673
Q375 655 313 605
Q153 480 153 279
Q153 186 190 141
Q208 119 226 119
Q242 119 263 137
Z"/>

</defs>

<!--example glyph image-->

```
<text x="0" y="1000" font-size="1000"
font-family="IPAMincho">GID13135</text>
</svg>
```

文字列「文字のテスト」の中の、「の」の部分に作成した SVG ファイルを配置した HTML ファイルを作成する。

[SVGFont.html]

```
<html>
  <head>
    <title>SVG Font</title>
  </head>
  <body>
    <p style="font-size: 50px;">
      文字<embed src="GID13135.svg" type="image/svg+xml"
width="50" height="50" />テスト
    </p>
    <p style="font-size: 75px;">
      文字<embed src="GID13135.svg" type="image/svg+xml"
width="75" height="75" />テスト
    </p>
  </body>
</html>
```

SVG フォント機能に対応している Web ブラウザとしては、Safari 4 以降、Opera 10 以降、Google Chrome 3 以降などがあり、その他のブラウザも順次対応予定となっている。

図 17 は、「SVGFont.html」を Web ブラウザ Safari4.0.1 で表示した結果である。「文字」と「テスト」部分はわかりやすいようにゴシックフォントを指定している。SVG フォントの「の」が文章中にきれいに並んで配置されることがわかる。

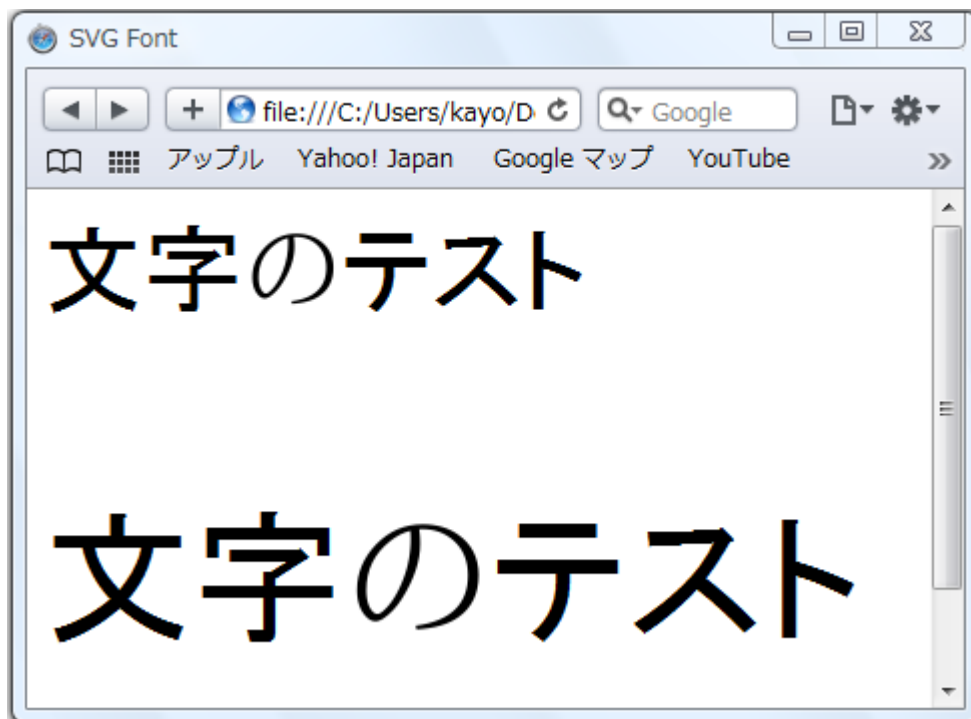


図 17 Safari4.0.1 での SVG フォント表示例

結論として、グリフデータベースに収納するグリフフォーマットを、SVG Tiny 1.2 準拠とすることにした。

5. 5 グリフデータベースの実装

「5. 4 グリフデータベースの提案」で検討した内容を元に、グリフデータベースを実装した。

実装に当たり留意した機能は、グリフの検索機能である。

グリフ検索者は、検索したいグリフの情報をどの程度もっているかわからない。コード番号や読みがわかっているグリフを検索する場合と、そのグリフの形しかわかっていないグリフを検索する場合は、必要となる検索機能が変わってくる。さまざまなレベルに対応するために、以下の5つの検索機能を実装した。

- 簡易検索
- コード検索
- 部首検索

- IDS 検索
- 登録情報から検索

(1) 簡易検索

簡易検索は、検索したいグリフの文字情報がある程度把握できている時の検索機能である。書体は必須とし、文字、画数、音読み、訓読み、読み方置換情報のいずれか1つ以上を入力することでグリフが検索できる（図 18）。

図 18 簡易検索

(2) コード検索

コード検索は、検索したいグリフのコード情報がある程度把握できている時の検索機能である。書体は必須とし、文字、ユニコード番号、JIS 面区点、康熙字典番号、大漢和辞典番号、Adobe Japan 1-6 番号のいずれか1つ以上を入力することでグリフが検索できる（図 19）。

簡易検索	コード検索	部首検索	IDS検索	登録情報から検索
<input type="button" value="検索する"/>				
書体 :	<input type="text" value="明朝"/>	※必須		
文字 :	<input type="text"/>	※IM によるテキスト入力		
開始Unicode : U+	<input type="text"/>	※16 進数4 桁～ 5 桁		
終了Unicode : U+	<input type="text"/>	※16 進数4 桁～ 5 桁		
JIS 面区点 :	<input type="text"/> - <input type="text"/> - <input type="text"/>			
康熙字典 :	<input type="text"/> . <input type="text"/>	※同文書局版のページを基本とした文字番号		
諸橋大漢和 :	<input type="text"/>			
Adobe Japan1-6 :	<input type="text"/>	※ CID 番号		
<input type="button" value="検索する"/>				

図 19 コード検索

(3) 部首検索

部首検索は、部首画数索引と部首名索引を選択できるようにした。

部首画数索引では、検索したいグリフを構成する部首の画数（1～17 画）を元に部首を特定することができるようにした（図 20）。部首名索引では、検索したいグリフを構成する部首の名前（50 音順）を元に部首を特定することができるようにした（図 21）。部首画数索引と部首名索引のどちらかにより部首を選択したうえで、グリフの検索を行うことができる。

書体： ※必須

検索対象部首： ※OR検索

1画	2画	3画	4画	5画
6画	7画	8画	9画	10画
11画	12画	13画	14画	15画
16画	17画			

候補部首：

図 20 部首検索（部首画数索引）

書体： ※必須

検索対象部首： ※OR検索

あ	か	さ	た	な
は	ま	や	ら	わ

部首の読み：

図 21 部首検索（部首名索引）

(4) IDS 検索

IDS 検索は、文字の構成を表す IDC と構成要素である Unicode で符号化されている漢字字形により検索する機能である (図 22)。

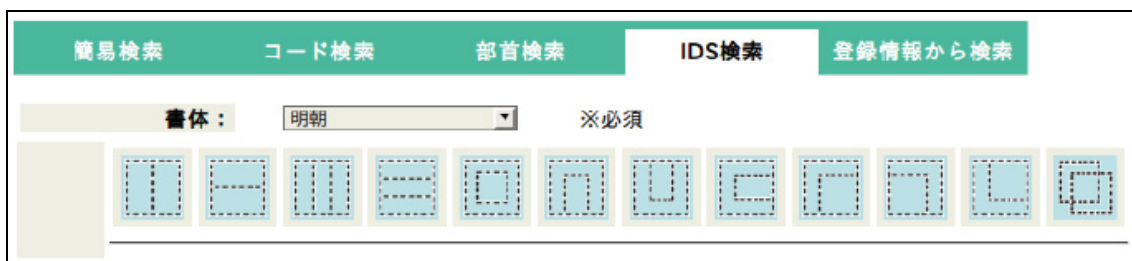


図 22 IDS 検索

専 (U+5C03) を検索する例を示す。専は図 23 に示すとおり、IDC (U+2FF1) と「甫」と「寸」で IDS 表記される。

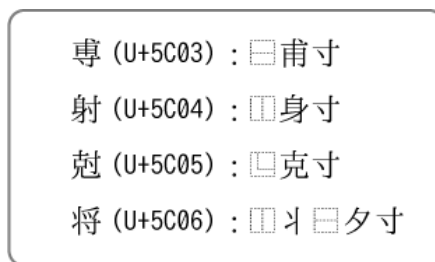


図 23 IDS の例

IVS 検索においては、図 24 のように IDC を可視的な構造図形より選択し、次に図 25 のように、IDC の次に構成される文字を選択し、IVS 検索を実行する。すると、図 26 のように IDC (U+2FF1) と「甫」で構成されるグリフ候補がリストアップされる。

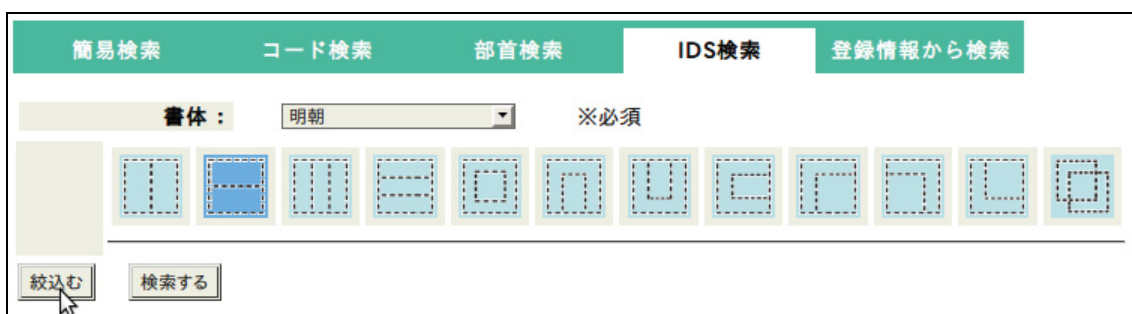


図 24 IVS 検索例 (1)

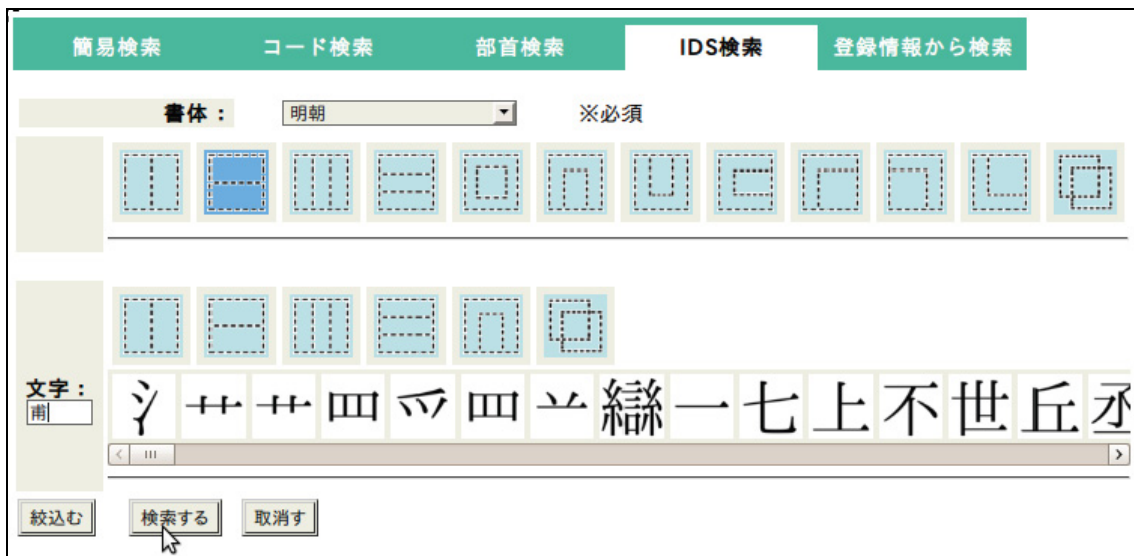


図 25 IVS 検索例 (2)

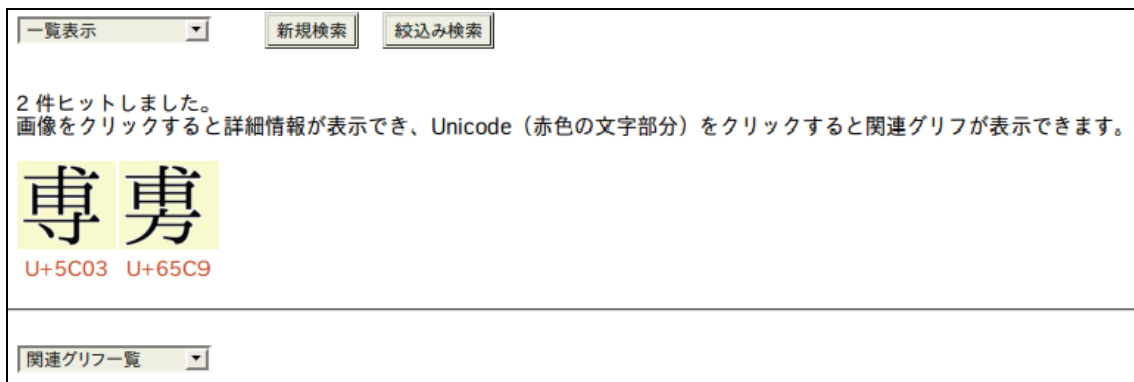


図 26 IVS 検索例 (3)

(5) 登録情報から検索

登録情報からの検索は、グリフ登録を行ったユーザーがグリフ検索を行う前提とした検索機能である。ユーザー名 (ID)、グリフ ID、備考欄のいずれか 1 つ以上を入力することでグリフが検索できる (図 27)。

簡易検索	コード検索	部首検索	IDS検索	登録情報から検索
<input type="button" value="検索する"/>				
書体 :	<input type="text" value="明朝"/>	※必須		
ユーザ-名 :	<input type="text"/>			
グリフId :	<input type="text"/>			
備考欄 :	<input type="text"/>	<input type="button" value="OR"/>		
<input type="button" value="検索する"/>				

図 27 登録情報から検索

5. 6 グリフデータベースの試用と評価

実装したグリフデータベースの試用を行った。

試用に際しては、データベースに登録するグリフを、IPA 明朝フォントおよび IPA ゴシックフォントに収納されているグリフで、各フォントにつき 12,209 グリフを登録した。

5. 6. 1 グリフデータベースの試用

試用において、以下の課題を実施した。試用は、IPA に所属する IPA フォント担当者 4 名を試用者とし、試用方法としては、与えられた同一の課題を 4 名それぞれが個別に実施するものとした。試用者の内の一名の操作についてはログを記録した。

(1) 課題

IPA 明朝フォントのうち、現在、文化庁審議が続いている新常用漢字⁵²の例示字形と異なる字形について 35 グリフ⁵³の変更を行う。その作業として、グリフデータベースから変更対象となるグリフを検索しダウンロードするとともに、グリフ変更
に有効なグリフパーツを検索しダウンロードすること。図 28 では、グリフ変更を

⁵²文化審議会国語分科会『新常用漢字表（仮称）に関する試案』（2009）
http://www.bunka.go.jp/oshirase_koubo_saiyou/2009/pdf/shin_kanjihyoshian.pdf

⁵³変更を行ったのは、IPA 明朝フォントと新常用漢字の例示字形が異なる次の 35 グリフである。「壓奥奥穀窮空姿資諮次就稱焼潜窓村沈突突備舞噴墳憤暴様林牙玩窟餌蹴拭蔑枕」

行う 35 グリフから 10 グリフを例示する。

NO.	IPA明朝	常用漢字
1	壓	壓
2	奥	奥
3	奥	奥
4	殼	殼
5	窮	窮
6	空	空
7	姿	姿
8	資	資
9	諮	諮
10	次	次

図 28 グリフ変更指示書

(2) 実施内容

NO.1 から順番にグリフの検索、ダウンロードを行った。

試用者 1 名の操作ログを表 14 に示す。なお、表中の No. はグリフ変更するグリフの通し番号 (図 28 に対応) であり、操作番号は No. と No. に対応したグリフの操作内容を枝番として付与した番号である。

表 14 グリフデータベースの試用結果

NO	操作番号	操作内容	結果	取得グリフ
1	1-1	部首検索(部首画数索引)で2画を選択	2画の部首が16部首リストアップされた	
	1-2	リストアップされた部首から「がんだれ」を選択し検索	20件ヒットしたが、所望のグリフが無かった。しかし修正パーツとして利用可能な「がんだれ」があったため、グリフをダウンロードした。	U+5382 (関連グリフ)
	1-3	部首検索(部首画数索引)で3画を選択	3画の部首が44部首リストアップされた。	
	1-4	リストアップされた部首から「つちへん」を選択し検索	188件ヒット、目視にて修正対象グリフを見つけてダウンロードした。	U+58D3
2	2-1	簡易検索で訓読み「おく」で検索	11件ヒットし、No.2およびNo.3の修正対象グリフを見つけた。No.2のグリフをダウンロードした。	U+5965
3	3-1	簡易検索で訓読み「おく」で検索	11件ヒットし、No.3のグリフをダウンロードした。	U+5967
4	4-1	簡易検索で訓読み「から」で検索	11件ヒットし、No.4のグリフをダウンロードした。同時に修正パーツとして使えそうな「殻」のグリフをダウンロードした。	U+6BBC U+6BBB (関連グリフ)
5	5-1	部首検索(部首画数索引)で5画を選択	3画の部首が27部首リストアップされた。	

	5-2	リストアップされた部首から「あなかんむり」を選択し検索	50件ヒット、No.5およびNo.6の修正対象グリフを見つけた。No.5のグリフをダウンロードした。	U+7AAE
6	6-1	部首検索(部首画数索引)で5画を選択	3画の部首が27部首リストアップされた。	
	6-2	リストアップされた部首から「あなかんむり」を選択し検索	50件ヒット、No.6のグリフをダウンロードした。	U+7A7A
7	7-1	部首検索(部首画数索引)で2画を選択	2画の部首が16部首リストアップされた	
	7-2	リストアップされた部首から「にすい」を選択し検索	31件ヒット、No.7のグリフは無かったが、修正パーツとして使えるような「にすい」のグリフをダウンロードした。	U+51AB
	7-3	簡易検索で訓読み「すがた」で検索	1件ヒットし、No.7のグリフをダウンロードした。	U+59FF
8	8-1	部首検索(部首名索引)でか行を選択	か行の部首が61部首リストアップされた・	
	8-2	リストアップされた部首から「かいへん こがい かい」を選択し検索	82件ヒット、目視にて修正対象グリフを見つけてダウンロードした。	U+8CC7
9	9-1	部首検索(部首画数索引)で7画を選択	7画の部首が21部首リストアップされた。	
	9-2	リストアップされた部首から「ごんべん」を選択し検索	222件ヒット。	

	9-3	絞り込み検索で、簡易検索で音読み「し」で検索	10件ヒット、目視にて修正対象グリフを見つけてダウンロードした。	U+8AEE
10	10-1	簡易検索で訓読み「つぎ」で検索	2件ヒットし、No.10のグリフをダウンロードした。	U+6B21

試用過程での、検索結果として、例えば図 29 は、簡易検索の訓読み「から」で検索した結果例である。No.4のグリフ (U+6BBC) および No.4の修正に使えるようなグリフ (U+6BBB) が同時に検索された。No.4の検索は1回の検索操作で所望のグリフにたどり着くことができた効率的な検索だったといえる。

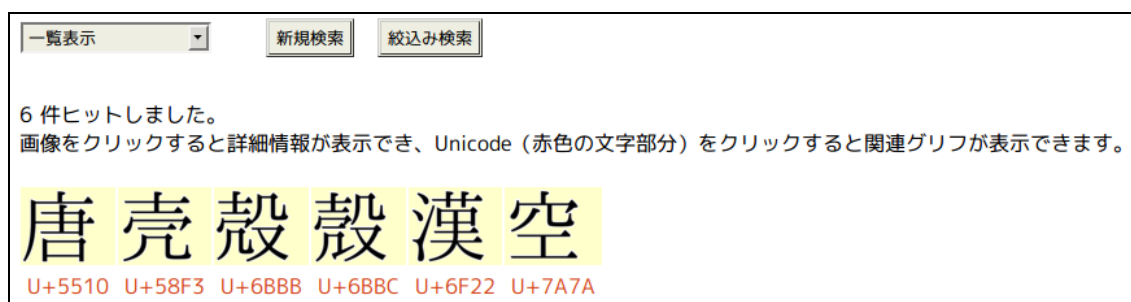


図 29 簡易検索 訓読み「から」で検索

図 30 は部首検索 (部首画数索引) 「にすい」で検索した例である。この中に NO.7,8,9,10 で変更する「にすい」の字形が存在する。しかし、NO.7,8,9,10 字体は検索結果には現れておらず、再度別の方法で検索を行っている。



図 30 部首検索（部首画数索引）「にすい」で検索した例

5. 6. 2 グリフデータベースの評価

試用者 4 名が同席の元、グリフデータベースの試用後の感想をヒアリングした。ヒアリングにより、得られた評価を以下に示す。

- ・ 検索方法については、最初からいくつかの選択肢があり、ユーザーの漢字の知識によって検索効率に差が出ることは間違いない。
- ・ 今回の試用では、最大 4 回の検索により所望のグリフまでたどり着くことができた。また、漢字の読みがわかる場合は 1 回の検索で所望のグリフにたどりついている。
- ・ 部首で検索する場合については、部首に対する知識がなければ、正確な画数や部首名はわからないので、いくつか部首検索を行ってみることで所望の部首を取得できる本システムはわかりやすいユーザーインターフェースであると言える。
- ・ グリフ検索において本システムは、複数の検索アプローチにより所望のグリフを引き出す能力を持っていると評価する。
- ・ 今回の試用では概ね問題なくグリフの検索を行うことができたが、さらに読みも部首も明確でない複雑な字形や近似するグリフが複数あるような場合に、知

識がなくても検索ができるようなユーザーインターフェースの工夫がほしい。

上記の評価はIPAに所属するIPAフォント担当者というある程度の文字に対する知見のある評価者の意見であるが、各人は検索の専門家ではなく、従って、本グリフデータベースが想定する「行政システム内で文字を検索するユーザー」の意見がある程度は代表するものである。しかしながら、今後、より優れた検索性を持つシステムとするためには、より多くの被験者による実験と検索ニーズの洗い出しが必要である。

5.7 おわりに

本章では、異体字、外字への対応の1つとして、グリフデータベースを提案し、実際に実装した。1万字を超える漢字は、フォントに収納されている漢字でさえ、InputMethodを使ってもアクセスできない場合がある。各官公庁、地方自治体を含めた行政システムの中で異体字、外字を扱う際も、所望のグリフがフォントに収納されているのか、あるいは異体字であるのか、新規に作成するしかないグリフなのか、の判断をすることさえ難しい状況である。本章で検証したグリフデータベースは、漢字情報としては十分なメタ情報を保有したシステムであり、検索性に優れている。課題としては、漢字の知識のないユーザーがグリフ検索を行うことのできるインターフェースである。例えば、手書き文字検索や、類似文字の違いを表現するインターフェースなどが考えられる。現在も、すでに手書き文字認識エンジンとして商用のアプリケーションやオープンソースプロジェクトがあるが、新規に登録した字形をカバーできるものが存在しない。オープンソースプロジェクトでは、Tomoe (Tegaki Online MOji-ninshiki Engine)⁵⁴があるが、手書き文字認識対象がJIS X 0208までの対応となっている。今後、こういったオープンソースプロジェクトを利用してJIS X 0213および異体字、外字への対応を行い、本グリフデータベースへ組み込んでいくことで、より優れたインターフェースを実現できると考える。

また、本グリフデータベースをオープンに公開することで、情報通信基盤として日本語フォントがどういった漢字を収納しているのかがわかると共に、漢字辞書としても利用できるし、外字利用の共通基盤としても利用出来ると考えている。

⁵⁴ <http://tomoe.sourceforge.jp/cgi-bin/ja/blog/index.rb>

第二部 多言語・多文字処理環境の整備のために

第六章 多言語文字処理環境の現状

6.1 はじめに

我が国における高度情報社会の発展のためには、ICT活用が必要不可欠であることは議論の余地はない。その実現においては、国境のないグローバル化と表裏一体として英語以外の言語を使った膨大な情報にいかにかに的確にアクセスし理解できるかが重要となってくる。

その際、多言語・多文字が情報アクセスへのバリアとなっている。現在のインターネット環境における使用言語は、英語以外の言語が60%を上回っているというMiniwatts Marketing Group(2009)による統計結果(図31)でも明らかなように、多言語による情報の発信と蓄積が増加している。さらに、Unicodeの整備などによりますますインターネットでの言語の多様化が進むことが予想される。

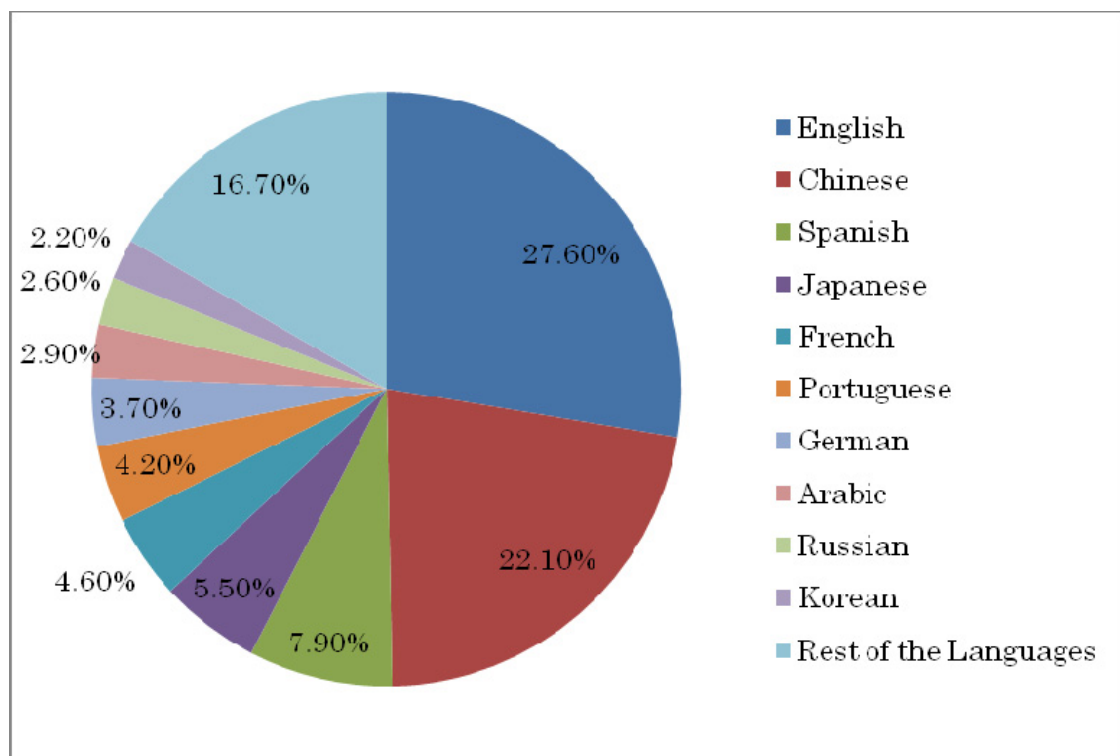


図 31 インターネット上で使用される言語のトップ 10
(Miniwatts Marketing Group(2009)のデータを元に筆者作成)

本章では、これまでの ICT における多言語文字処理環境を概括し、インターネットでの言語の多様化に対して我が国が取り組むべき多言語文字処理の課題について整理する。

6. 2 ICT における多言語環境の現状

第一部で日本語における文字処理環境を見てきたが、多言語文字処理についても同様にその言語とそれを記述する文字の数だけ、文字処理環境が必要となってくる。

とりわけ、英語等のラテン文字圏に比べ、インド系文字、アラビア文字、漢字については、扱う文字の量と文字の質の点でさまざまな文字処理環境の工夫が必要となる。ここで言う文字の質とは、たとえばアラビア文字であれば、左から右に文字組みされることや、1つの文字が語頭、語中、語末、独立で最大4つの字形変化をする(図32)ことなど変形、音節合成、合字、アクセントマークなどの非常に複雑な要素である。

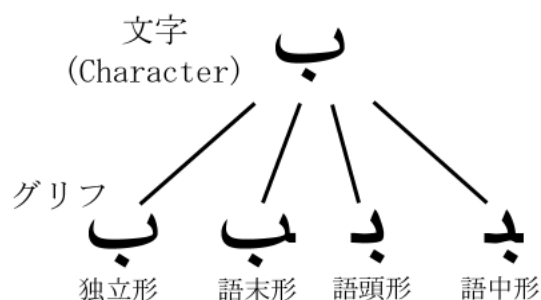


図 32 アラビア文字の字形変化

これまで、日本同様に世界においても、国または同一の言語・文字圏で独自の方式により、コンピュータによる文字処理がすすめられてきた。1つの国において、統一した文字コードが無く、メーカー主導により文字コードもフォントもキー配列も編集ソフトウェアもすべてセットとなっており他のプログラムとまったく互換性が無いという場合もあった。

例えば、筆者は、財団法人国際情報化協力センター (CICC) の多言語情報セキュリティ関連情報・収集分析システム開発事業において「文字コード体系及び文字表現に関する調査(2002)」として、カンボジア語を記述するクメール文字環境の調査および Unicode 符号化を前提とした InputMethod とフォントの開発を行った。

当時カンボジア国内において、カンボジア語で使われるクメール文字の情報交換用文字コードに関する国家標準がなく、コンピュータの導入に伴って販売会社が独自に文字コードを定めたフォントを作成したため、互換性のない文字コード体系が非常に多く存在していることがわかった。入力方法についても各社独自のキーボード配列により、キー対文字コードが1対1の入力となっていた。従って、2002年当時、カンボジア語による文字情報交換は事実上おこなえず、インターネットプロバイダでも、メールのやり取りやホームページには英語の使用を推奨していた。

インド系文字に属するクメール文字は、音節文字と呼ばれ、文字は子音文字と母音文字（母音記号）から成り、両者の組み合わせで子音と母音からなる音節単位を表す。さらにクメール文字には脚とよばれるクメール文字特有の特徴がある。これは子音の一種であるが、子音が2つ3つと重なる場合、その第2、第3の子音が脚となって、変化するものである。クメール文字は、変形、音節合成、合字、アクセントマークなどの非常に複雑な要素をすべて持つスクリプトの1つである。このクメール文字の複雑性と、過去の内戦等によるカンボジア国内の混沌による文字文化継承者不足やIT化の遅れが相まって、Unicode規格化においても、クメール文字符号化上の問題が発生していた。

小林（2005）が経緯の詳細を報告しているが、ユニコード規格では、母語使用者の参画なしに標準化されたスクリプトが存在するが、クメール文字についてもカンボジアの国家代表者の参画なしに決められた。その中で、カンボジアの文字文化にとっては非常に重要な脚文字が符号化されていないなどのユニコード符号化方式の技術的な欠陥がカンボジア政府およびそれを支援する代表団から提出された。この際、クメール文字の符号化を全面的にやり直すべきであるという主張であったが、一旦国際規格として決定した符号位置を変更することは困難であるため、主張の一部を受け入れ、不要な符号位置の使用を抑制するコメントの追加と必要な符号位置を追加および脚文字を表す符号列を明記し（図 33）、その符号位置に一意の名前を付与する改訂をUnicode 3.2で行った⁵⁵。しかし、カンボジアサイドでは、Unicode 3.2で定義されている脚文字処理や、クメール文字特有の音節結合文字の処理についての十分な技術的知見を持ち合わせていなかった。このため、問題がUnicode規格側

⁵⁵ Unicode3.2 khmer の追加記述

http://www.unicode.org/reports/tr28/tr28-3.html#9_15_khmer

にあるのか、フォント実装技術にあるのか、入力方法にあるのか混在してしまい、より混乱をもたらすこととなった。結果としては、CICC の調査プロジェクトで、Unicode 3.2 を元にしたクメール文字のフォント実装と InputMethod 実装により、クメール文字出力が可能であることが実証され、カンボジア政府の合意を得ることができた。

Subscript Characters									
័	៑	្	៓	។	៕	៖	ៗ	៘	៙
17D2 1780	17D2 1781	17D2 1782	17D2 1783	17D2 1784	17D2 1785	17D2 1786	17D2 1787	17D2 1788	17D2 1789
័	៑	្	៓	។	៕	៖	ៗ	៘	៙
17D2 178A	17D2 178B	17D2 178C	17D2 178D	17D2 178E	17D2 178F	17D2 1790	17D2 1791	17D2 1792	17D2 1793
័	៑	្	៓	។	៕	៖	ៗ	៘	៙
17D2 1794	17D2 1795	17D2 1796	17D2 1797	17D2 1798	17D2 1799	17D2 179A	17D2 179B	17D2 179C	17D2 179D
័	៑	្	៓	។	៕	៖			
17D2 179E	17D2 179F	17D2 17A0	17D2 17A2	17D2 17A7	17D2 17AB	17D2 17AF			

図 33 Unicode 3.2 で追記された脚文字を表す符号列

この例からわかるように、多言語文字処理において、特に発展途上国では、Unicode 等の標準規格への参加が費用面で困難であったり、IT 技術的に遅れていたため、自国だけでは満足のいく規格定義が不可能な場合がある。そこには、文化的、政治的な問題により、情報通信環境においてラテン文字圏とは比較にならないデジタル・ディバイドを生み出す要因が存在していた。アジアの中の先進国である日本としては、アジア諸国の IT 化を援助する意味でも、国際整合性、相互運用性を配慮した多言語文字処理に対する知見を深める必要がある。

美馬(2001)によると、デジタル・ディバイドとは、米国商務省が 1998 年に発行

したレポートが語源とされ、米国商務省は「一方に、最も優れたコンピュータや、最も優れた電話サービス、最も速いインターネットサービスを利用できる人々があり、もう一方に何かしらの理由でそれらを利用できない人々がいる場合、これらの人々の間における違いをデジタル・ディバイドと言う。」としている。我が国では、旧郵政省の『21世紀の情報通信ビジョン－IT JAPAN for ALL－』の中で、「所得、年齢、教育レベル、地理的要因、身体的制約要因等によるインターネット等の情報通信手段に対するアクセス機会及び情報通信技術を習得する機会の不平等（デジタル情報格差（Digital Divide））が国内及び国家（地域）間で顕在化しており、こうした機会を持つ者と持たざる者との格差が拡大しつつあるとの指摘がなされている。」と定義しており、他にも近藤(2000)、木村（2001）によりデジタル・ディバイドが狭義と広義に定義されている。本論文が示す多言語におけるデジタル・ディバイドとは、米国商務省が示す定義に基づく。

一方で、インターネットでの言語の多様化は、デジタル・ディバイドを解消するチャンスの到来ともいえる。Unicodeにより各言語・文字圏でのばらばらな文字コードが世界共通となり、OpenType フォントというオープンなフォントフォーマットが公開されたことで、多言語文字処理の壁は低くなったと言える。

2009年10月1日には、「The Unicode Standard, Version 5.2.0」がリリースされ、Bamum、Javanese、Lisu、Meetei Mayek、Samaritan、Tai Tham、Tai Vietの7スクリプトの新規追加と既存スクリプトの文字追加、合わせて6,648文字が追加された。Unicodeはメジャー語だけでなく少数民族や途上国をもサポートしつつある。

しかし、Unicode化されさえすれば、文字処理環境のすべてが効率的に整うわけではない。第一部でも述べたように、文字処理環境には、入力、処理、出力の環境が整わなければならない。

6. 3 多言語文字処理研究の状況

6. 3. 1 多言語間の連携

我が国で、最も研究が盛んな多言語文字処理分野は翻訳技術であろう。総務省によるUNS戦略プログラムの「ユニバーサル・コミュニケーション技術戦略」の中でも、「スーパー・コミュニケーション」として音声翻訳技術やテキスト翻訳技術が

期待されている（情報通信審議会答申、2007:104）ように、言語の壁を翻訳技術により取り払うことが、ユニバーサル・コミュニケーションの課題とされている。

機械翻訳には、変換方式（トランスファ方式）と中間言語方式（ピボット方式）がある。変換方式（トランスファ方式）では、まず原言語で書かれた入力文を解析して、原言語の文法に沿った構文構造を得たのち、変換規則を用いて目標言語の構文構造に変換し、目標言語の文を生成する。中間言語方式（ピボット方式）では、入力文をより深く解析し、言語に依存しない中間言語で記述された表現に変換し、その表現から目標言語の文を生成する。これらには、文法規則、単語辞書、意味辞書など様々なデータが必要となり、これまでに多くに研究が進められてきた。

しかし、一方で機械翻訳の問題も指摘されている。

宮部、吉野、重野(2009)による、多言語医療受付支援システムの構築では、外国人患者との対話において、医療分野では極めて高い翻訳精度が要求されており、機械翻訳技術による低精度な翻訳による不十分な意思疎通が重大な問題を引き起こすことがある、としており、用例対訳を用いタッチパネルによる質問形式により、コミュニケーションを図っている。用例対訳とは、多言語話者によりあらかじめ正確に翻訳された多言語の対訳を指す。

機械翻訳に対する問題としては、稲葉ら(2009)による機械翻訳を用いた3言語間コミュニケーションの相互理解の分析において、機械翻訳を用いた場合に自分と他者、あるいは他者と他者のコミュニケーションを正しく理解できないため、共通基盤の成立が困難である、としている。

機械翻訳の品質には未だ十分ではないという前提でNICT等が研究推進している「言語グリッド⁵⁶」は、多言語サービス基盤としてインターネット上の言語資源（対訳辞書、シソーラス、コーパス）や言語処理機能（言語の解析、機械翻訳、言い換え）をセマンティック Web の技術により連携されることにより、世界規模の連携で言語の壁を越える試みである。各国からの言語資源を蓄積することで、各国の言語サービスが充実する事となるが、現在カバーしている言語は欧米各国の言語と、中国語、韓国語、タイ語などの比較的 IT 化が早くから進んでいるアジア地域の言語に限られている。

⁵⁶ 言語グリッドとは：<http://langrid.nict.go.jp/jp/project.html>

これら、翻訳技術を見た場合、研究対象の多くは、英語、中国語、韓国語などの限られた言語であり、また入力、出力部分の文字処理については、特に意識されていない。

いずれの研究も、すでにある多言語情報資源をどのように解析処理するか、という命題に対する研究であり、その情報資源にいかにかアクセスするか、という単純な問題に対するアプローチがこれまで存在しなかった。情報資源にいかにかアクセスするかとは、すなわち、多言語文字入力を行い、検索し、出力することである。

6. 3. 2 多言語文字入出力

多言語を「あらゆる言語」と置いた場合、1つのコンピュータ上で、どのように文字を入力するか、と言う問題がたちはだかる。日本人のコンピュータ環境を前提とした場合、日本語入力と欧文入力はさほど困難ではないが、その他の言語を入力しようとするとならば、たちまち面倒な OS レベルのインストール作業が必要となる。

文字入力方法としては、ラテン文字やインド系文字、アラビア文字などの表音文字系のスクリプトにおいては、キーボードに文字コードが直接アサインされている直接入力方式がとられている。キー配列としては、ラテン文字では、英字の QWERTY 配列がデファクトスタンダードとなっているが、その他の言語においても、ドイツ語圏・チェコ語圏では QWERTZ 配列、フランス語圏では AZERTY 配列などがある。さらに異なるスクリプトにおいては、それぞれの文字に対応したキーボード配列が必要となる。従って、所望の文字入力を行うためには、その文字に対応したキーボード配列を規定し、それを変更する手段を OS レベルで実装している必要がある。

一方、日本語や中国語、韓国語などの多文字圏では、InputMethod（あるいは FEP：Front-End-Processor と呼ぶ）というソフトウェアが必要となる。これは、1. キーボード上で入力した文字列を変換辞書を用いて解釈し、候補文字の一覧を画面表示し、2. ユーザーが表示候補の中から、その1つを選択する、という処理である（Ken Lunde,1999）。

日本語における InputMethod の技術は、森ら（1978）による「かな漢字変換」による一連の処理として、入力したい文章の読みをキーボードから入力し、意味・文法解析を行い、読みと漢字との変換対応表（かな漢字変換辞書）を用いて、漢字

仮名交じり文に変換する、という処理の精緻化が進められてきた。現在は、高度な InputMethod として、OS 標準搭載では、WindowsOS に搭載されている Microsoft IME (Input Method Editor) または Microsoft Office IME、Mac OS のことえりがあり、OSS で開発されている iBUS、IIIMF、SCIM (Smart Common Input Method)、uim や、商用のソフトウェアとしてジャストシステム(株)の ATOK などがある。

さらに、日本語の入力環境については、田中ら (2003) による携帯等の少数のキーでの入力の研究、高林ら(2002)、市村ら(2002)や佐藤ら(2006)による入力の変換効率の向上や予測変換機能などの研究が続けられている。また、清田ら (2007) による視覚障害者の日本語文字入力支援システムや、山口ら (2007) による視線移動を利用した肢体不自由者に向けた応用可能な入力支援システム方法の研究も行われている。日本語の入力環境は一定の水準に達しているといえよう。

しかしながら、これら日本語入力環境を多言語にまで応用する研究はほとんど見受けられない。

現実的には、直接入力方式または InputMethod による変換入力方式のいずれも、どの言語 (スクリプト) を入力できるかは、使用 OS 環境の設定に依るところが大きい。従って、新たに日本語や英語以外の言語をいままでの OS 環境のまま利用しようとする、言語 (スクリプト) によっては、相当の困難が付きまとう。

一方、文字入力を画像から行う研究としては、OCR による文字認識技術や手書き文字認識技術の研究が進められてきた。これらの研究は、森 (1972) の郵便番号の読み取りやタイプライターといった特定用途の文字認識の研究から、堀ら(2001)の TV 映像内テロップ文字や、松尾ら(1997)のカメラ撮影した情景画像に含まれる文字にまで研究領域が拡大し、その認識精度の向上が図られている。多言語を対象とした研究では、高橋ら (2005) によるデジタルカメラで撮影した画像中のハングルを対象とした文字列領域の抽出などが進められている。

仮に入力環境があったとしても、その言語 (スクリプト) で入力した文字列が可視的な文字として正しく表示されるためには、出力環境として、その言語 (スクリプト) に対応したフォントとレンダリングが必要である。日本語フォントについてはこれまで、田中ら (1995) や上地 (2002) 等によりフォントの自動作成技術の研究などが進められてきたが、多言語フォントについては調査レベル (鈴木ら、2006)

の研究に留まっている。多言語フォントやレンダリング品質などの出力環境については、OS レベルに依るところが大きく、研究としてはあまり多く存在しない状況にある。

6. 4 多言語文字処理の現状と課題

世界的に OS シェアの 90%以上を確保する WindowsOS (図 34) においては、Microsoft が多言語対応を積極的におこなっていることもあり、OS レベルでの対応言語が着実に増えている。



図 34 Operating System Market Share(2009,10)⁵⁷

2009 年 10 月に発売が開始された Windows7 では、アラビア語、ブラジル系ポルトガル語、ブルガリア語、繁体中国語 (香港)、簡体中国語、クロアチア語、チェコ語、デンマーク語、オランダ語、英語、エストニア語、フィンランド語、フランス語、ドイツ語、ギリシャ語、ヘブライ語、ヒンディー語、ハンガリー語、イタリア語、日本語、韓国語、リトアニア語、ノルウェー語、ポーランド語、ポルトガル語、ルーマニア語、ロシア語、セルビア語 (ラテン)、スロバキア語、スペイン語、スウェーデン語、タイ語、トルコ語、ウクライナ語の 35 言語に対応している⁵⁸。また、インド系を中心に 50 のフォントを新たに追加している⁵⁹。上記の 35 言語を見ても、ヒンディー語、タイ語以外のインド系文字には対応していない。従って、その他の言語は入力さえまならない状況に置かれている。

⁵⁷

<http://marketshare.hitslink.com/operating-system-market-share.aspx?qprid=8>

⁵⁸ <http://www.microsoft.com/japan/athome/umall/win7/function/05.aspx#02>

⁵⁹ <http://blogs.msdn.com/e7jp/>

現在の入力環境で、キーボード上あるいは InputMethod を使って入力することができない文字があった場合は、例えば WindowsOS が提供している MSIME では図 35 の IME パッドのような文字一覧から文字を探しだし、入力する手段を準備しているが、母語使用者にとっても、その言語を熟知しない使用者にとっても非常に入力しにくい環境であることは間違いない。日本語入力においても、InputMethod が変換候補として挙げてこない漢字をこういったユーザーインターフェースで探すこともあるが、ある言語のすべての文字をこのインターフェースで入力することは負荷以外の何者でもない。

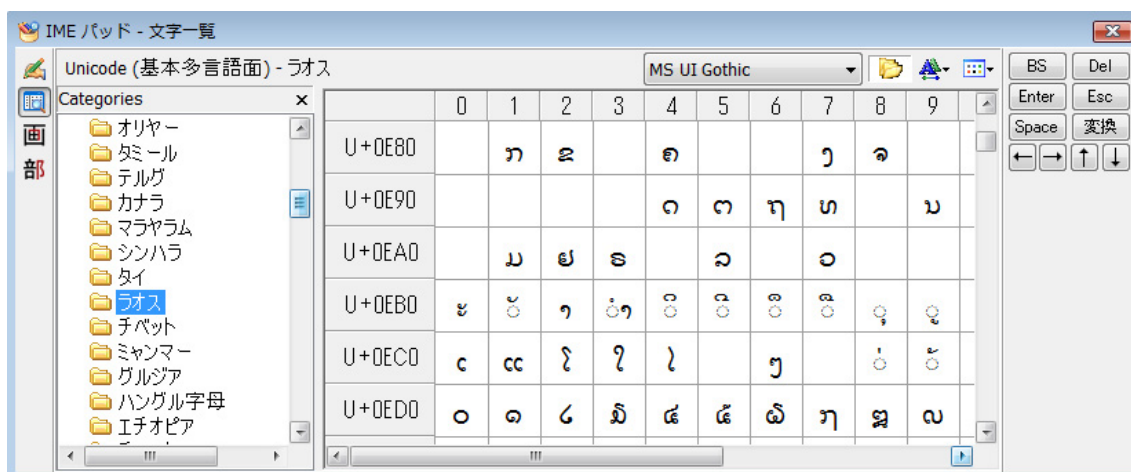


図 35 IME パッドでの文字一覧

また、「対応している」といった場合にもさまざまなレベルが想定され、常に入出力環境の調整が必要となる。例えば、2009年10月1日に制定されたばかりの「The Unicode Standard, Version 5.2.0」で追加された文字を例にする。Devanagari (U+0900~U+097F) に U+097B~U+097F の文字が追加された (図 36)。

Sindhi implosives		
097B	𑖀	DEVANAGARI LETTER GGA
097C	𑖁	DEVANAGARI LETTER JJA
Glottal stop		
097D	𑖂	DEVANAGARI LETTER GLOTTAL STOP
		<ul style="list-style-type: none"> • used for writing Limbu in Devanagari • a glyph variant has the connecting top bar
Sindhi implosives		
097E	𑖃	DEVANAGARI LETTER DDDA
097F	𑖄	DEVANAGARI LETTER BBA

図 36 The Unicode Standard Version 5.2.0 による Devanagari の追加

Devanagari はヒンディー語、マラーティー語の表記に使用される他、Sindhi 語を表記するためにも使われるが、これまでの Unicode の Devanagari 領域のコードテーブルには Sindhi 語に必要な図 36 の文字が入っていなかった。

文字コードとして定義された後に、完全にその文字が所望の形で出力されるためには、まずフォントの対応が必要である。そして、その文字コードを入力するための入力方法が必要となる。ここまでが整ってもさらにもう一つ重要な要素がレンダリングである。とくにインド系文字の場合、子音字 (consonant) と母音記号 (vowel) から文字を形成するため、レンダリングがそのスクリプトの文字構成に対応しないと所望の形での出力ができない場合がある。WindowsOS の場合、「Uniscribe⁶⁰」が Unicode によって符号化されたテキストを描画するためのレンダリングサービスであり、「Uniscribe」は「USP10.DLL」というダイナミックリンクライブラリとして実装されている。「USP10.DLL」は、OS のバージョンや OS のアップデートによって異なるバージョンが組み込まれており、たとえ同じ WindowsXP だとしても、「USP10.DLL」のバージョンが異なると、文字表示に違いが出る場合がある。上記 Devanagari の追加文字（例では U+097B=DEVANAGARI LETTER GGA）に対する対応についても、最新の USP10.DLL Ver.1.626.7100.0 では問題ないが、少し古いバージョンではうまくいかない。図 37 と図 38 は全く同じ文字列であるが、

⁶⁰ <http://msdn.microsoft.com/en-us/library/dd374091%28VS.85%29.aspx>

USP10.DLL のバージョンが 1.471.4063.0 の図 38 では、子音字に下に母音記号が配置されているが、古い「USP10.DLL」の場合、図 38 のように、以前からあった右上の文字については、子音字に下に母音記号が配置されるが、右下の「The Unicode Standard, Version 5.2.0」で追加された文字については、子音字と母音記号が分かれて表示されてしまう。

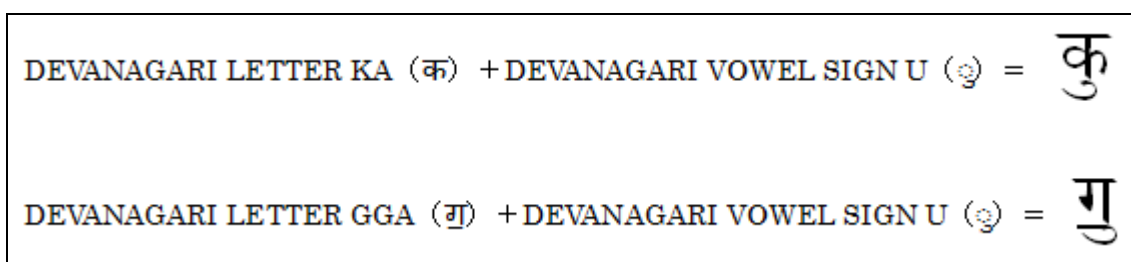


図 37 Devanagari のレンダリングが出来ている例
(USP10.DLL Ver.1.626.7100.0)

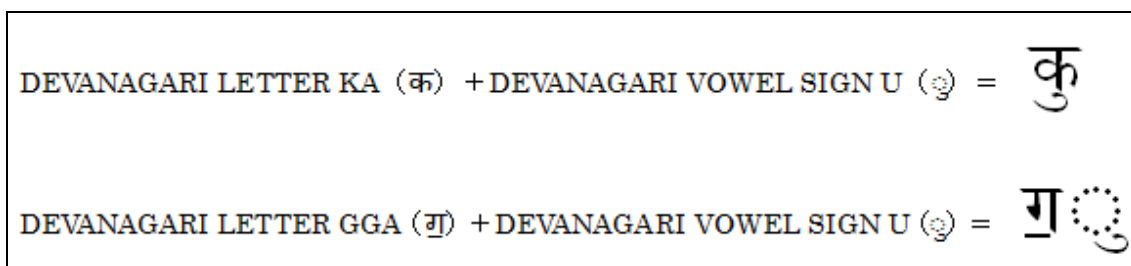


図 38 Devanagari のレンダリングが出来ていない例
(USP10.DLL Ver. 1.471.4063.0)

このように、多言語文字処理には、入出力環境に複数の対応要件が控えているため、多言語対応の問題については、どこが現状でボトルネックになっているのかを見極める必要がある。

フォントやレンダリングといった出力環境は OS 側に依るところが多いため、本研究では、現状把握にとどめる事とする。

多言語文字処理の必要性について考える場合、多くのユーザーにとって必要なことは、母語や得意とする言語以外では、その言語での情報を検索し、表示できる環境が整っていることである。さらに、その言語を母語や得意とする言語に自動翻訳

できれば、情報理解や言語学習にとって有益であろう。

しかし、多くのユーザーにとって、ラテン文字と母語以外の文字を入力する環境を整備し、実際に文字入力をするには非常に難しい状況にあることもわかった。さらに、国際化する中、海外のインターネットカフェ等のオープンな環境で、Web上の検索を行う際にも、検索するための文字が入力できない、という言語の障壁が立ちはだかる。

これまで、文字入力は、ローカルホストの OS レベルに依存しており、多言語文字入力を行う場合には、その言語用の入力方法のインストールやキーボード配列の選択が必要であり、また、多くの言語の場合は、キーボード配列に支配された入力を余儀無くされていた。

そこで、第七章では、Unicode 符号化によりテキスト化されたあらゆる多言語情報資源へのアクセス方法として、ユーザー環境に縛られないインターネットブラウザでの入力操作を実現する方法を提案する。

第七章 多言語 InputMethod の実装と評価実験

7.1 はじめに

本研究では、利用したい言語に不慣れな言語学習途上のユーザーでも簡単に所望の言語の Web 上の情報資源にアクセスできるような、多言語に対応した入力支援システムを提案する。

7.2 問題の所在と目的

これまでの文字入力環境は、ユーザーのローカルホスト上のキーボード配列や InputMethod の設定に依存しており、多言語を入力するためには、OS レベルでの入力環境の設定が必要であった。さらに、OS レベルでの入力方法の提供がない言語の場合は、独自にその言語用の入力方法を準備しなくてはならず、多言語の文字入力環境の設定は容易とは言えない状況にある。このようなローカルホストの入力環境からの脱却として、Web 上の InputMethod が挙げられる。横山ら(2004)によるブラウザ上でインターネットを介して日本語入力ができる日本語 Web IME の研究のように、海外においてローカルホストに日本語入力環境がなくても日本語入力が行えるような研究が進められている。横山ら(2004)の研究は、Web 上の図書館蔵書情報の検索のための入力支援システムであるが、同様に Web 上に公開されているあらゆる言語の情報資源へのアクセスを考えた場合、ローカルホストに各言語の入力環境がなくても所望の言語を入力する手段が求められるが、これまで多言語をターゲットとした Web 上の InputMethod の研究は存在しなかった。

Web 上に公開された情報資源へのアクセスは、主に Web ブラウザでの操作により行われる。Web ブラウザでの情報資源へのアクセス方法は複数あり、一つは特定の Web ページへの直接的なアクセスであり、この場合は既知の URL へのリンクをクリックすることや、URL を直接入力するなど所望の情報資源へのアクセスが可能となる。もう一つには、Google や Yahoo! に代表されるポータルサイトと呼ばれるインターネットの入り口となる Web サイトを経由したアクセスであり、ポータルサイトでは、検索エンジンや Web ディレクトリなどのサービスを展開している。こ

のうち Web ディレクトリは、人知によって分野別に分類した Web サイトの索引集であり、分野が階層構造になっていることから、所望の分野の階層を辿ることで Web サイトへのアクセスが可能となる。ただし、索引集に登録されていない Web サイトや、分野が特定できない Web サイトへのアクセスは不可能である。一方、不特定の Web サイトを対象とした検索エンジンの中で、Google のようなロボット型検索エンジンでは、クローラーと呼ばれるプログラムが周期的に全世界の Web コンテンツを取得し自動的に検索用のインデックスを DB に収納する。クローラーが Web コンテンツを収集する範囲や、インデックスの作り方は各検索エンジンで異なるものの、ロボット型検索エンジンに対し、さまざまな言語での検索クエリー（キーワード）を指定することでさまざまな言語の情報資源へのアクセスが可能となってきた。

しかしながら、各検索エンジンで指定する検索クエリーの文字入力の問題となる。

広大な Web 空間に多様な多言語情報資源が蓄積されていても、Web 検索によりそこにアクセスする手段が簡単には手に入らない状況である。そこで本研究では、情報資源の対象をインターネットで Web 公開されている Unicode (UTF-8) で記述されたあらゆる言語データと置き、そのデータへアクセスするために必要な検索エンジンで指定する検索クエリーを入力するための多言語 InputMethod を提案し、実装、評価を行う。

提案には、ユーザー対象を「利用したい言語に不慣れな言語学習途上のユーザー」と置くことで、だれもが簡単に多言語入力を行える環境の提案を目指す。また、ユーザーがその言語を入力するためにユーザーの PC (クライアント) での特別なセットアップの必要が無いことを前提としたシステム提案を行う。

7. 3 提案システム

本研究では、クライアント側の OS 環境に依存せず、多言語情報資源から情報検索するために、対象言語の制約を持たない文字入力手段の提供を目的とするため、Internet Explorer などのインターネットブラウザで入力操作を行える次のようなシステムを提案する。

7. 3. 1 多言語対応

多言語対応の対象を Unicode で規格化されているすべての言語 (スクリプト) と

する。キーボード上のキーにすべての文字コードをアサインすることは不可能であることは言うまでもない。ここには、多文字を処理してきた日本語 InputMethod の処理工程の 1 つである「辞書との照合」というプロセスを持ち込む。辞書とは、入力文字列と出力文字列を対応させる変換辞書である。

入力文字列には、世界中のどのようなパソコン環境でも間違いなく入力可能な ASCII 領域内の文字（制御コードは含まない、図 39 参照）を利用する。従って、キーボード上のキーに入力文字列をすべてアサインすることができる。

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0020		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
0030	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0040	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0050	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
0060	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
0070	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

図 39 ASCII 領域内入力文字一覧

出力文字列は、UTF-8 エンコードの文字列とし、Unicode 化されているどのような言語（Script）でも出力可能とする。

変換辞書は、各言語（スクリプト）毎に入力文字列と出力文字列の対応を示す変換辞書を準備する。

本研究で、各言語における入力文字を ASCII 領域内の文字とするという考え方は言語学における転写（transcription）を活用したものである。転写とは、言語の音声を一定の規則に基づいて文字表記することをいう。これまで、インド系文字やアラビア文字などでは、主に ASCII 領域内の文字を利用して転写表記したもの（ローマ字転写ともいう）を情報資源として利用してきた（町田、2001）。日本語のヘボン式や訓令式といったローマ字表記も一種の転写規則とみることができる。

さまざまな言語の転写規則については、規格化されたものがあるわけではなく、各言語の研究者が音声とソーティングなどの利便性の面で工夫した転写規則を各自で定めている。従って 1 言語について、何通りもの転写規則を作ることは可能となる

が、本研究では、多くの言語に対応することができることを実証することが目的であるので、転写規則の詳細については議論しない。

変換辞書は、転写規則を元に、入力文字列を ASCII 領域内の文字とし、出力文字列を各言語の UTF-8 エンコードの文字列とした対応表であり、言語毎の変換辞書を工夫することにより、すべての言語(スクリプト)に本システムは対応可能となる。

7. 3. 2 語彙辞書

変換辞書に登録する入力文字と出力文字の対応は、直接入力と同じ結果を期待する場合は、入力文字:出力文字=1文字:1文字であるが、日本語等の InputMethod による変換入力に似た結果を期待する場合は、入力文字:出力文字= n 文字: m 文字($n \geq 1, m \geq 1$)とすることができる。

後者では語彙を変換辞書に投入することを前提としており、本研究での提案は、次に示すインクリメンタルサーチにより語彙の綴りをうろ覚えのユーザーに配慮した設計とした。

7. 3. 3 インクリメンタルサーチ

入力しようとしている言語に不慣れなユーザー向けに入力を支援する手段として、インクリメンタルサーチでの文字変換を行う。

インクリメンタルサーチとは、ユーザーが1文字入力する度に候補文字を表示していくこと検索手法で、逐語検索、逐次検索とも言う。Jef Raskin(2000)は、インクリメンタルサーチにより検索がすばやく行えるだけでなく一打鍵毎にユーザーにフィードバックが返る点が優れている、と主張している。

インクリメンタルサーチは、これまでも GNU プロジェクトによるテキストエディタの Emacs⁶¹や、高林ら(2002)による日本語のインクリメンタル検索手法である「Migemo」や、携帯電話や Google の検索エンジンに利用されている。

これまでの日本語の InputMethod のような変換辞書を伴う文字変換では、一連のキー入力を終了後、文字変換を指示することで初めて変換候補がリストアップされた。

インクリメンタルサーチでは、最初の1文字(C_1)を打鍵した段階で、 C_1 を先頭

⁶¹ <http://www.gnu.org/software/emacs/emacs.html>

に含む変換候補がリストアップされる。さらに続けて 1 文字 (C₂) を打鍵すると、C₁C₂ を先頭に含む変換候補がリストアップされる。

これにより、完全に入力文字列を打鍵すること無しに、所望の語彙に絞り込まれる。特に、入力しようとしている言語に不慣れなユーザーは、その文字の綴りに自信がないことが想定されるため、インクリメンタルサーチにより、一打鍵毎のユーザーへのフィードバックにより候補文字が表示されることは、正確な綴りに早く導かれるきっかけをあたえるものであると仮説設定する。

7. 3. 4 詳細情報表示 (支援機能)

本研究の新規性の 1 つは詳細情報表示にある。

本研究では、ユーザー対象を「利用したい言語に不慣れな言語学習途上のユーザー」とおいている。このターゲットユーザーは、その言語における入力したい語彙のつづりに自信がなかったり、類似したつづりの単語の区別がつかなくなったりすることが想定される。そこで、インクリメンタルサーチにより入力しようとしている語彙が本当に入力したい語彙であることを確認する手段として、変換候補となる語彙に関する情報、例えば語彙の意味の説明や日本語、英語などの翻訳、文例など、を表示する詳細情報表示機能を提案する。

詳細情報は変換辞書内に、入力文字列と出力文字列とともに収納する。インクリメンタルサーチにより、出力文字列が変換候補としてリストアップ表示された際に、詳細情報を表示する。詳細情報は Web ブラウザで表示することを前提としているため、データ記述には HTML タグを利用可能とし、詳細情報の表示には多言語 (多スクリプト) はもとより音声や画像の提示や他の Web ページへのリンクも行えることとする。

詳細情報を充実させることで、その言語に不慣れなユーザーが間違いなく所望の語彙入力ができ、かつ言語学習の補助としての機能を保有することを仮説設定する。

7. 4 システム実装

7. 4. 1 システム構成

Ajax (Asynchronous JavaScript + XML) 技術を用いてシステム実装を行った。開発環境には、非同期通信処理の実現手段として Google Web Toolkit (GWT) を

使用し、ミドルウェアとして Apache2.0.59、Apache Tomcat 5.5.26、PHP5.2.2、phpMyAdmin2.10.1 を利用した。

本提案システムで Ajax を用いる理由としては、本提案システムが、ユーザーにローカルホストへのプラグインソフトのインストール等の環境設定における負荷を与えないことを目指しており、ユーザー環境に Web ブラウザのみがあればよい環境を実現するためである。また、従来の Web ブラウザを使った Web アプリケーションでは、データをサーバに通知して処理結果を得るにはページ全体をロードしなおさなければならず、本提案システムのようなインクリメンタルサーチを伴う入力方法を実現することは不可能であったが、非同期通信を利用し、通信結果に応じてダイナミック HTML で動的にページの一部を書き換える Ajax の技術により、実現可能となった。

各言語（スクリプト）の入力に必要な変換辞書データは、リレーショナル DB である MySQL 5.0.82 に収納した。

7. 4. 2 システムの特徴

通常のサーバー・クライアントシステムでは、HTML フォームや URL パラメータといったデータをサーバに送信し、サーバはそのデータに基づいた処理を行い、結果を含めた HTML ページを自動作成してクライアントに返し、表示する、という処理スタイルをとる。本システムでは、Ajax の非同期通信処理を利用しており、ページ遷移を一切伴わずに、クライアント・サーバー間の通信を実現している。

また、本システムでは、HTML ページのエンコード、Java で記述するクライアント・サーバーサイドのソースコード、DB のデータのすべてを UTF-8 でエンコードすることによりあらゆる言語（スクリプト）に対応し、ブラウザ上に多言語を表示することを可能としている。入力文字列として使用される文字は、ASCII 領域内の文字である。

本システムでは、インクリメンタルサーチにより、入力された文字列を元に DB 検索を行い、変換候補をリストアップ表示する。出力文字の確定はユーザー操作により行われるか、入力文字列に対し変換候補となる出力文字列が 1 語も存在しない場合は、一文字前にさかのぼって検索を行い、変換候補が存在するところで、自動的に変換が行われる。

また、本システムは、Web 空間への情報検索を目的としているため、検索クエリーには、スペースを区切りとした複数語の入力が求められる。一方、多言語対応においては、多くの分かち書きを行う言語の変換辞書収納が課題となる。システムとして両者を両立させるために、スペース（U+0020）の扱いが問題となる。本システム実装では、変換辞書に登録する入力文字列にはスペースを使用することは不可とし、分かち書き言語で複数語からなる熟語等を1つの出力文字列として登録したい場合は、スペースを“_”(アンダーライン、LOW LINE：U+005F) とする規則を設けた。

7. 4. 3 変換辞書データ

本システムのデータベースには、変換辞書データを言語毎にテーブルとして収納した。各テーブルは、入力文字列を search_term (varchar⁶²(100))、出力文字列を term(vchar(100))、詳細辞書を info (text⁶³)として設計した。図 40 にヒンディー語でのテーブル例を示す。

search_term	term	info
aMDaa	अंदा	अंदा<...
aMtaRii	अंतड़ी	अंतड़...
aMd'aa	अंधा	अंधा<...
aMd'aapana	अंधापन	अंधाप...
Ad'eraa	अंधेरा	अंधेर...
aMtaHpura	अंतःपुर	अंतःप...
Ad'aurii	अंधौरी	अंधौर...
akaala_1	अकाल ?	अकाल ...
akaala_2	अकाल २	अकाल ...
akelaa	अकेला	अकेला...
akk'aRa	अक्खड़	अक्खड़...
aktuubara	अक्तूबर	अक्तू...

図 40 ヒンディー語のテーブル例

⁶² Verchar は MySQL におけるデータ型の1つで可変長の文字列であることを示す。Verchar(100)とした場合は、100文字までを1つのセルに収納することができる。

⁶³ text は、任意の長さの文字列を収納することができるデータ型である。

実際に登録する変換辞書データは、簡易な作成・登録をめざし、以下の手順で作成し登録を行う。

(1) オリジナル辞書データの作成

入力文字列、出力文字列、詳細情報をタブ区切りし、改行を 1 列の区切りコードとした UTF-8 のテキストデータファイルとして作成する。

本システムにとって、オリジナル辞書データが重要であることに相違ない。オリジナル辞書データは、入力文字列、出力文字列のみでも登録が可能である。詳細情報については、文字数制限はなく、HTML タグを使用することができる。ただし、詳細情報内のテキストとしてタブと改行の使用は不可とした。

本システムの実装テストに使用するオリジナル辞書データは、三省堂辞書シリーズや東京外国語大学アジア・アフリカ言語文化研究所の研究者が保有する各言語の語彙辞書を元にして本システム用のオリジナル辞書データ形式に変換したデータである。

これら既存データの多くには、カタカナ発音または転写が含まれているので、output string から key input string を生成することは比較的容易に行うことができる。

(2) SQL データの生成

本システムに合わせ、MySQL への登録用に (1) を SQL データに変換するプログラムを Java で開発した。この変換プログラムにより、(1) から SQL ファイルを生成する。

(3) MySQL への登録

Web ブラウザ上の phpMyAdmin の操作により、(2) で作成した SQL ファイルを本システムに登録する。

7. 4. 4 ユーザーインターフェース

クライアントは、Web Browser 上で入力操作をおこない、文字確定後、検索サイトに文字列を検索クエリーとして投げることができる。

一連の入力操作を以下に示す。

(1) 入力言語の選択

Web Browser 上のプルダウンメニューにより、入力したい言語を選択する (図 41)。

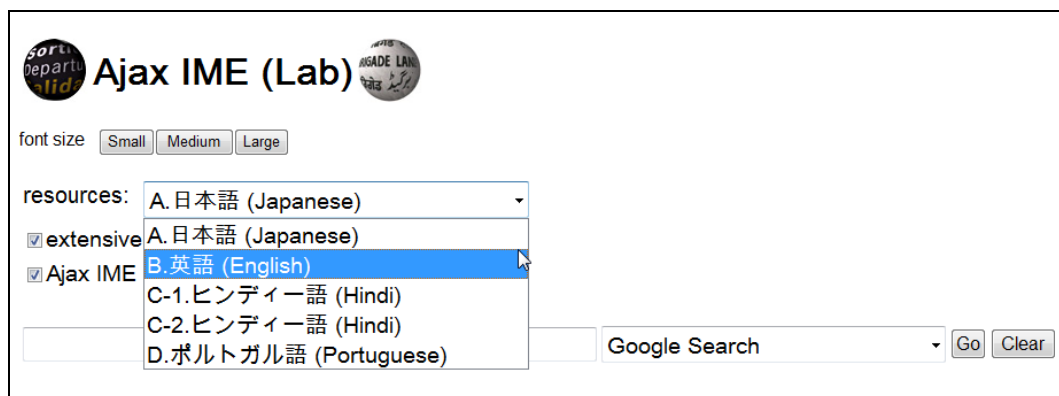


図 41 入力言語の選択

(2) インクリメンタルサーチによる文字入力

ここでは、Devanagari を Script として利用するヒンディー語の入力例を示す。

サンプルとして登録したヒンディー語用の変換辞書には、合計 7,524 語からなる東京外国語大学アジア・アフリカ言語文化研究所のヒンディー語辞典を使用した。

ヒンディー語の「Book: 本」は「किताब」であり (町田、2008)、変換辞書内の入力文字列は「kitaaba」である。

一連の入力操作を図 42 に示す。

まず、「kitaaba」の先頭文字「k」を入力する。変換辞書より「k」が先頭文字となる出力文字列候補がリストアップされる。変換辞書に登録されている語のうち「k」から始まる語は 555 語であり、kitaaba は 449 番目である。ここで、リストアップされる出力文字列候補の順列は、文字コードによるソーティングの昇順である。

次に、「i」を入力すると、変換辞書より「ki」が先頭文字となる出力文字列候補がリストアップされる。変換辞書に登録されている語のうち「ki」から始まる語は 44 語であり、kitaaba は 41 番目である。

次に「t」を入力すると、変換辞書より「kit」が先頭文字となる出力文字列候補がリストアップされる。変換辞書に登録されている語のうち「kit」から始まる語は

2語だけなので、ここで「किताब」を確定する。あるいは、確定操作として、マウスで出力文字列候補をクリックするか、キー操作により出力文字列候補を選択状態にし、エンターキーを打鍵するか、のいずれかである。

The diagram illustrates the step-by-step input of the Hindi word 'किताब' (kitāb). It is divided into three stages: 'k', 'ki', and 'kit'. Arrows at the top indicate the flow from 'k' to 'ki' and then to 'kit'. Each stage shows a list of candidates on the left and a detailed information box on the right. The selected candidate in each stage is highlighted in blue, and its details are shown in a yellow box.

“k” を入力	“i” を入力	“t” を入力																																																																										
<table border="1"> <tr><td>k</td><td></td></tr> <tr><td>क्षमा</td><td>k\$amaa</td></tr> <tr><td>क्षमता</td><td>क्षमा f.</td></tr> <tr><td>क्षण</td><td>pardon. forgiveness IHED: p.037</td></tr> <tr><td>क्षत्रिय</td><td>vt. क्षमा करना</td></tr> <tr><td>क्षेत्र</td><td></td></tr> <tr><td>क्षेत्रफल</td><td>*मैं क्षमा मांगता हूँ I beg your pardon. IHED: p.037</td></tr> <tr><td>ख</td><td></td></tr> <tr><td>खाई</td><td>*उसे क्षमा करना . Please forgive him. IHED: p.037</td></tr> <tr><td>खादी</td><td></td></tr> <tr><td>खाल</td><td></td></tr> <tr><td>खाली</td><td>क्षमा f. pardon, forgiveness TEACH YOURSELF</td></tr> <tr><td>खामोशी</td><td></td></tr> <tr><td>खान</td><td></td></tr> <tr><td>खान-पान</td><td></td></tr> <tr><td>खाना ?</td><td></td></tr> </table>	k		क्षमा	k\$amaa	क्षमता	क्षमा f.	क्षण	pardon. forgiveness IHED: p.037	क्षत्रिय	vt. क्षमा करना	क्षेत्र		क्षेत्रफल	*मैं क्षमा मांगता हूँ I beg your pardon. IHED: p.037	ख		खाई	*उसे क्षमा करना . Please forgive him. IHED: p.037	खादी		खाल		खाली	क्षमा f. pardon, forgiveness TEACH YOURSELF	खामोशी		खान		खान-पान		खाना ?		<table border="1"> <tr><td>ki</td><td></td></tr> <tr><td>कि</td><td>ki</td></tr> <tr><td>किधर</td><td>कि.</td></tr> <tr><td>किफ़ायत</td><td>that (as conjunct TEACH YOURS</td></tr> <tr><td>कीचड़</td><td></td></tr> <tr><td>कील</td><td></td></tr> <tr><td>कीमत</td><td></td></tr> <tr><td>कीड़ा</td><td></td></tr> <tr><td>कीर्तन</td><td></td></tr> <tr><td>कीर्ति</td><td></td></tr> <tr><td>की बग़ल में</td><td></td></tr> <tr><td>की हैसियत से</td><td></td></tr> <tr><td>की ओर</td><td></td></tr> <tr><td>की तरफ़</td><td></td></tr> <tr><td>की तरह</td><td></td></tr> <tr><td>की तुलना में</td><td></td></tr> </table>	ki		कि	ki	किधर	कि.	किफ़ायत	that (as conjunct TEACH YOURS	कीचड़		कील		कीमत		कीड़ा		कीर्तन		कीर्ति		की बग़ल में		की हैसियत से		की ओर		की तरफ़		की तरह		की तुलना में		<table border="1"> <tr><td>kit</td><td></td></tr> <tr><td>किताब</td><td>kitaaba</td></tr> <tr><td>कितना</td><td>किताब f. book. IHED: p.032 syn. पुस्तक</td></tr> <tr><td></td><td>*हिंदी की किताब . book of Hindi. IHED: p.032</td></tr> <tr><td></td><td>किताब f. book TEACH YOURSELF:</td></tr> </table>	kit		किताब	kitaaba	कितना	किताब f. book. IHED: p.032 syn. पुस्तक		*हिंदी की किताब . book of Hindi. IHED: p.032		किताब f. book TEACH YOURSELF:
k																																																																												
क्षमा	k\$amaa																																																																											
क्षमता	क्षमा f.																																																																											
क्षण	pardon. forgiveness IHED: p.037																																																																											
क्षत्रिय	vt. क्षमा करना																																																																											
क्षेत्र																																																																												
क्षेत्रफल	*मैं क्षमा मांगता हूँ I beg your pardon. IHED: p.037																																																																											
ख																																																																												
खाई	*उसे क्षमा करना . Please forgive him. IHED: p.037																																																																											
खादी																																																																												
खाल																																																																												
खाली	क्षमा f. pardon, forgiveness TEACH YOURSELF																																																																											
खामोशी																																																																												
खान																																																																												
खान-पान																																																																												
खाना ?																																																																												
ki																																																																												
कि	ki																																																																											
किधर	कि.																																																																											
किफ़ायत	that (as conjunct TEACH YOURS																																																																											
कीचड़																																																																												
कील																																																																												
कीमत																																																																												
कीड़ा																																																																												
कीर्तन																																																																												
कीर्ति																																																																												
की बग़ल में																																																																												
की हैसियत से																																																																												
की ओर																																																																												
की तरफ़																																																																												
की तरह																																																																												
की तुलना में																																																																												
kit																																																																												
किताब	kitaaba																																																																											
कितना	किताब f. book. IHED: p.032 syn. पुस्तक																																																																											
	*हिंदी की किताब . book of Hindi. IHED: p.032																																																																											
	किताब f. book TEACH YOURSELF:																																																																											

図 42 ヒンディー語入力例

詳細情報は図 42 の背景色が黄色になっている部分である。選択状態（背景色が青色）の出力文字列に対応する詳細情報が表示される。図 42 の例では、HTML 記述により色指定、フォント指定を行っている。

(3) 検索エンジンへの遷移

入力確定した文字列を最終的には、Web への検索クエリーとして利用する。本システムでは、Go ボタンをクリックすることで、検索エンジン（Google で実装）に検索クエリーを引き渡すところまでを一連の操作として構築した（図 43）。



図 43 検索エンジンへの遷移

7. 4. 5 クライアント・サーバー間の通信

「7. 4. 4 ユーザーインターフェース」で示したユーザー操作に対応したクライアント・サーバー間の通信を図 44 に示す。

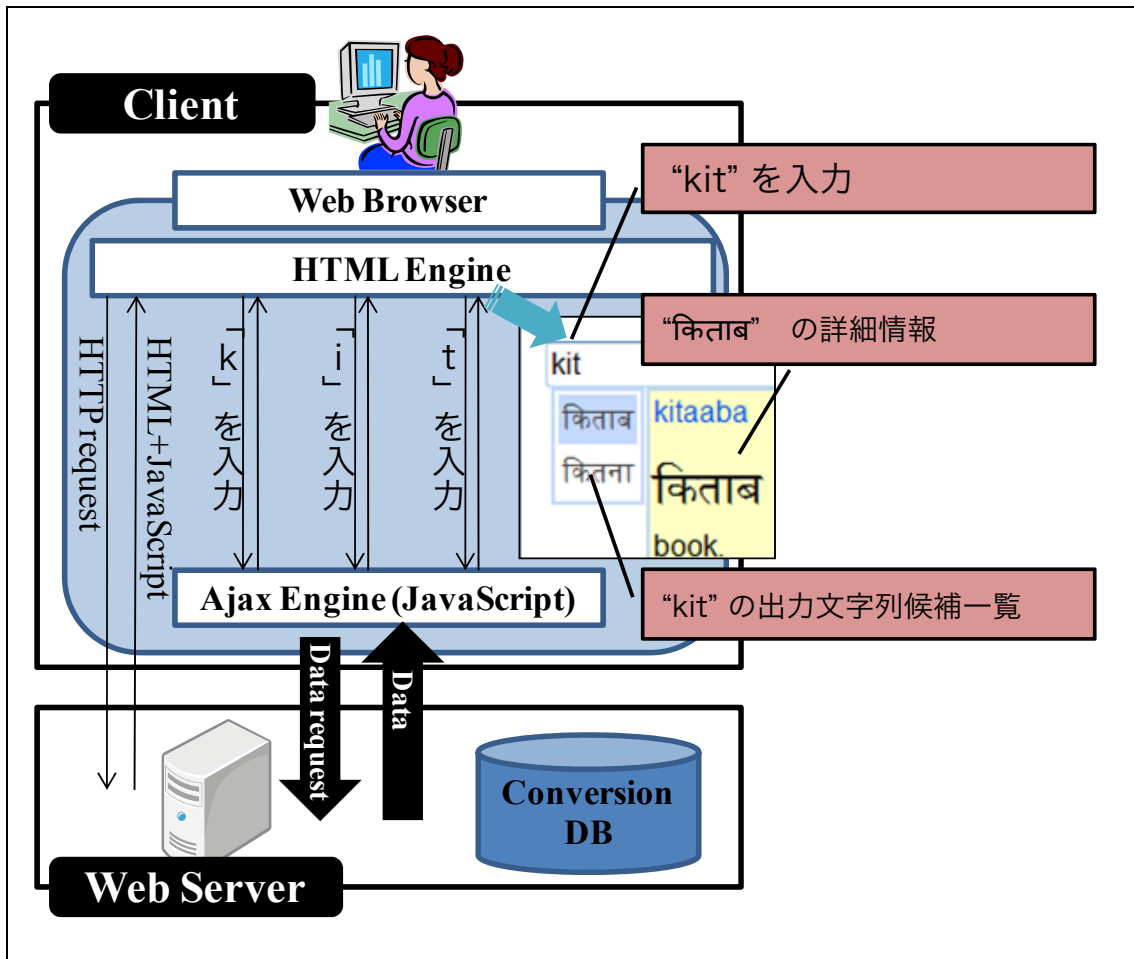


図 44 クライアント・サーバー間の通信

7. 5 評価

実装したシステムには、インド系文字 10 言語 (Devanagari, Tamil, Telugu, Thai, Bengali, Gurmukhi, Kannada, Sinhala, Sanskrit, Malayalam)、アラビア文字 3 言語 (Persian, Uyghur, Urdu)、日本語、英語、ポルトガル語の変換辞書を登録し、Web 画面上のプルダウンメニューで言語を切り替えることで、所望の言語の入力を行うことができた。

さらに実装したシステムを使って、「7. 3 提案システム」での仮説設定を評価するための実験を行った。

7. 5. 1 ユーザーによる入力実験

本システムをユーザーに使用してもらい評価実験を行った。実験は、「7. 3 提案システム」での以下の仮説設定を評価することを目的とする。

- ① インクリメンタル検索を用いた本システムにより、その言語に不慣れなユーザーが所望の語彙を正確に早く入力できる。
- ② 詳細情報により、その言語に不慣れなユーザーが正確に所望の語彙入力でき、かつ言語学習の補助としての機能を保有する。

なお、以下評価実験の中では、本システムを「AjaxIME」と呼ぶ。

(1) 方法

クライアント環境において、Windows Vista OS 上の Internet Explorer 8 を使い、言語毎に、「既存の入力方法」と「AjaxIME」により、解答用紙により提示されたあらかじめ評価用に準備した文字列を入力してもらい、その文字列を検索クエリーとして検索エンジン（Google）で検索した際の結果を解答用紙に記録してもらう。検索結果には検索件数を用いる。本番問題を始める前に、入力方式毎の操作方法の説明および練習問題を5問行ったのち、制限時間を10分として30問の検索と検索結果（検索件数）の記録を行う（図45）。

「既存の入力方法」と「AjaxIME」での入力実験が終了したのち、ユーザーには2つの方法の操作についての主観評価をヒアリングした。



図 45 入力実験方法

また、本実験では、本番問題でのユーザーのキーボードによる文字入力、文字入力以外のキー操作（スペースキー、エンターキー等の打鍵）、マウス操作をすべて記録する。

本実験では、結果として、各入力方式での回答数、正答数、誤答数を集計する。回答数とは、検索結果を回答できた設問の数である。正答数、誤答数は、ユーザーが記録した検索件数と、実験当日に実際に正しく文字列を入力した際の検索件数とが一致した場合を正答、一致しなかった場合を誤答として集計される設問数である。

なお、実験当日に実際に正しく文字列を入力した際の検索件数は、同じネットワーク上でも若干の誤差が生じる場合があるため、検索件数が実際に正しく文字列を入力した際の検索件数と異なる場合は、ユーザーのキー操作の記録をもとに正答か誤答かの判断を行った。

（２）実験対象言語と既存の入力方法および AjaxIME の設定と被験者

日本語、英語、ヒンディー語、ポルトガル語を実験対象として既存の入力方法と AjaxIME の比較を行う。4 言語を対象とする理由としては、本研究の提案方式が多数の言語に対応していることを示すとともに、ユーザーの慣れが異なる言語において既存の入力方法と AjaxIME の評価を行うことで、その言語に不慣れなユーザーにとって AjaxIME が有効である仮説設定を確認することである。そこで、①使い慣れた日本語、②比較的入力が容易であり義務教育レベルである程度の知識を持つ英語、③使用する文字が特殊（Devanagari 文字）で学習経験がないと入力がほとんどできないヒンディー語、④基本はラテン文字であるが学習経験がなく綴りになじみがないポルトガル語、というそれぞれ異なる要素を持つ 4 言語での文字入力実験を行うこととした。

また、詳細辞書が有効であるという仮説設定を確認するため、ヒンディー語では、既存の入力方法と AjaxIME での入力に加え、AjaxIME の詳細情報を表示しない「詳細情報なし」についても同時に実験する。

言語毎の「既存の入力方法」および「AjaxIME」で利用する辞書データと実験用に準備した入力文字列は以下の通りである。

① 日本語

「既存の入力方法」としては、OS 標準の Microsoft IME を用いる。

「AjaxIME」で利用する辞書データとして、三省堂大辞林データを元にした約 2 万語を使用する。三省堂大辞林データに含まれる情報は、出力文字列と詳細情報にあたる。入力文字列は、出力文字列のカタカナ読みを元にヘボン式で変換した文字列である。

実験用に準備した入力文字列は、上記三省堂大辞林データから乱数発生によりランダムに抽出した 30 語である。

被験者は、H01~H06、P01~P06 までの合計 12 名であり、全員が同一の「既存の入力方法」と「AjaxIME」の入力比較テストを実施した。

② 英語

「既存の入力方法」としては、キーボードによる直接入力を用いる。

「AjaxIME」で利用する変換辞書データとして、三省堂デイリー辞典のデータを元にした約 1 万 5 千語を使用する。三省堂デイリー辞典データに含まれる情報は、出力文字列と詳細情報にあたる。入力文字列は、出力文字列とほぼ同じ文字列であるが、複数語で構成される場合にはスペースを“_”(アンダーライン)に変換する。

実験用に準備した入力文字列は、上記三省堂デイリー辞典から乱数発生によりランダムに抽出した 30 語である。

被験者は、H01~H06、P01~P06 までの合計 12 名であり、全員が同一の「既存の入力方法」と「AjaxIME」の入力比較テストを実施した。

③ ヒンディー語

「既存の入力方法」としては、オープンソースソフトウェアとして公開されている Virtual Keyboard v3.5.3⁶⁴を用いる。ヒンディー語の入力方法は、日本語 OS 環境に標準的には搭載されていない。そこで、本実験では、バーチャルキーボードとして、モニター上にキーボード配列を表示し、バーチャルキーボードをマウスクリックまたはハードウェアとしてキーボードを打鍵することで各言語の入力が可能な Virtual Keyboard v3.5.3 を用いた。ヒンディー語を母語として利用する場合の入力

⁶⁴ <http://debugger.ru/projects/virtualkeyboard>

はキーボードにすべての文字が割り当てられている直接入力方式であり、Virtual Keyboard v3.5.3 では、ヒンディー語専用のハードウェアとしてキーボードなしにヒンディー語の入力が可能となる。



図 46 Virtual Keyboard v3.5.3

「AjaxIME」で利用する辞書データとして、東京外国語大学の所有する 7,524 語が収納されたヒンディー語辞書を使用する。ヒンディー語辞書データに含まれる情報は、出力文字列と詳細情報にあたる。入力文字列は、転写規則をもとに出力文字列から生成した。本実験で使用した転写規則は必ずしも万人に共通するものではない。したがって、「AjaxIME」ユーザーに対しては、転写規則表をあらかじめ提示しておく。

ヒンディー語については他の言語同様に詳細情報を含む辞書に加え、詳細情報を含まない辞書を準備し、それぞれを「詳細辞書あり」、「詳細辞書なし」として入力実験をおこなった。

実験用に準備した入力文字列は、上記ヒンディー語辞書から乱数発生によりランダムに抽出した 30 語である。

被験者は、H01~H06 までの合計 6 名であり、全員に同一の「既存の入力方法」と「AjaxIME」の入力比較テストを実施した。ヒンディー語を記述する Devanagari 文字はラテン文字とくらべて誰でもすぐにわかる文字ではないため、ある程度のヒンディー語の知識が必要である。本実験では、ヒンディー語の学習経験のある 6 名を被験者とした。ヒンディー語レベルの自己評価として、1. 自由に使える 2. 読

み書きができる 3. 読むことができる 4. なんとか読むことができる 5. 勉強したことがあるだけ、の 5 段階選択方式をとった結果、2 が 3 名、3 が 2 名、4 が 1 名であり、いずれも自由に使えるレベルには達していない。

④ ポルトガル語

「既存の入力方法」としては、オープンソースソフトウェアとして公開されている Virtual Keyboard v3.5.3 を用いる。ポルトガル語は ASCII 文字内のラテン文字とポルトガル語特有の記号で構成されるが、その記号を入力するためには、OS レベルでの入力方式切り替えが必要となるため、ヒンディー語と同じ Virtual Keyboard v3.5.3 を用いる。ポルトガル語を母語として利用する場合の入力はキーボードにすべての文字が割り当てられている直接入力方式であり、Virtual Keyboard v3.5.3 では、ポルトガル語専用のハードウェアとしてキーボードなしにポルトガル語が入力可能となる

「AjaxIME」で利用する辞書データとして、三省堂デイリー辞典のデータを元にした約 1 万 5 千語を使用する。三省堂デイリー辞典データに含まれる情報は、出力文字列と詳細情報にあたる。入力文字列は、出力文字列とほぼ同じ文字列であるが、ポルトガル語特有の記号は（例：extraordinário）、入力文字列からは削除し（例：extraordinario）、複数語で構成される場合にはスペースを“_”(アンダーライン)に変換している。

実験用に準備した入力文字列は、上記三省堂デイリー辞典から乱数発生によりランダムに抽出した 30 語である。

被験者は、P01~P06 までの合計 6 名であり、全員がポルトガル語をまったく知らないメンバーとし、全員に同一の「既存の入力方法」と「AjaxIME」の入力比較テストを実施した。

以上の 4 言語に関する被験者実験にそれぞれ実験 No.を付与し、表にまとめたものが表 15 である。

表 15 被験者実験の実験 No.と内容

実験 No.	日本語		英語		ヒンディー語			ポルトガル語	
	A1	A2	B1	B2	C1	C2	C3	D1	D2
入力方法	既存の入力方法 : Microsoft IME	Ajax IME	既存の入力方法 : 直接入力	Ajax IME	既存の入力方法 : Virtual Keyboard v3.5.3	Ajax IME (詳細情報なし)	Ajax IME (詳細情報あり)	既存の入力方法 : Virtual Keyboard v3.5.3	Ajax IME
AjaxIMEで利用する変換辞書データ	-	三省堂大辞林データ約2万語	-	三省堂デイリー辞典データ約1万5千語	-	東京外国語大学ヒンディー語辞書7,524語		-	三省堂デイリー辞典データ約1万5千語
被験者	12名				6名			6名	

7. 5. 2 結果

(1) 回答と操作数による評価

既存の入力方式と本研究が提案する「AjaxIME」方式における被験者の回答と操作数の平均比較を行い、日本語、英語、ポルトガル語の結果に対しては、「同一の被験者」が「両方式を操作」する評価テスト方式のため、平均値の差の検定「対応のある t 検定 (両側検定)」を実施した。

ヒンディー語については、「従来方式」、「AjaxIME 詳細情報なし」、「AjaxIME 詳細情報あり」の 3 方式に対する被験者の回答と操作数の平均比較を行うため、「同一の被験者」が 3 方式を操作する「対応のある一元配置の分散分析 (3つの平均値の差の検定)」を実施した。また、「対応のある一元配置の分散分析」の結果、有意差がみられた要因については、多重比較(Tukey 法)による分析⁶⁵を行った。

(a) 正答数

既存の入力方式と AjaxIME の正答数の平均を比較した (表 16、表 17)。

結果として、日本語 (T=0.073, n.s.)、英語(T=4.91, n.s.)では有意な差はなかった。

ポルトガル語において、「既存の入力方式 (D1)」と「AjaxIME (D2)」を比較した場合、「既存の入力方式」は正答数の平均 23.17、「AjaxIME」の正答数の平均が

⁶⁵ 分析には、javascript-star

(<http://www.kisnet.or.jp/nappa/software/star/puma/as.htm>) を使用した。

28.17 であり、「AjaxIME」の正答数の平均が 2.50 多く有意（両側検定）な差を付けた（ $T=0.046$, $p<.05$ ）。まったく知識の無い言語においても、「AjaxIME」が「既存の入力方式」よりも早く正確に文字入力ができることが明らかとなり、設定した仮説を確認できた。

ヒンディー語において、3 方式の平均の分散分析の結果、有意な差がみられた（ $p<.01$ ）。「既存の入力方式（C1）」、「AjaxIME 詳細情報なし（C2）」、「AjaxIME 詳細情報あり（C3）」それぞれのペアにおける有意差を検出するために、回答率に対する多重比較(Tukey 法)による分析を行った。

「既存の入力方式」と「AjaxIME 詳細情報なし」および、「既存の入力方式」と「AjaxIME 詳細情報あり」においては有意な差が見られた（共に $p<.05$ ）。しかしながら「AjaxIME 詳細情報なし」と「AjaxIME 詳細情報あり」においては有意な差は見られなかった。

従って、ヒンディー語において、「AjaxIME」が「既存の入力方式」よりも早く正確に文字入力ができることが明らかとなり、設定した仮説を確認できた。一方、詳細情報の有無が入力に与える影響については、確認することができなかった。

表 16 正答数（対応のある t 検定（両側検定））

被験者	日本語		英語		ポルトガル語	
	正答数A1	正答数A2	正答数B1	正答数B2	正答数D1	正答数D2
H01	29	27	27	30	-	-
H02	29	29	29	29	-	-
H03	29	25	30	30	-	-
H04	29	29	29	30	-	-
H05	29	29	30	30	-	-
H06	30	27	30	30	-	-
P01	29	26	30	23	10	24
P02	27	30	30	30	28	30
P03	29	27	30	28	26	29
P04	24	25	28	29	27	30
P05	27	26	29	30	20	26
P06	22	19	30	26	28	30
平均	27.75	26.58	29.33	28.75	23.17	28.17
平均値の差 (t検定)	0.073		0.461		0.046	
5%水準*					*	
1%水準**						

表 17 正答数（対応のある一元配置の分散分析）

被験者	ヒンディ語		
	正答数C1	正答数C2	正答数C3
H01	12	28	24
H02	11	30	29
H03	23	21	26
H04	17	23	24
H05	10	20	25
H06	10	28	29
P01	-	-	-
P02	-	-	-
P03	-	-	-
P04	-	-	-
P05	-	-	-
P06	-	-	-
平均	13.83	25.00	26.17
一元配置の 分散分析	0.007		
5%水準*	**		
1%水準**			

(b) 誤答数

「既存の入力方式」と「AjaxIME」の誤答数の平均を比較した（表 18、表 19）。

日本語、英語においては、誤答数の有意な差は見られなかった。

ポルトガル語において、「既存の入力方式（D1）」と「AjaxIME（D2）」を比較した場合、「既存の入力方式」は誤答数の平均が 1.33、「AjaxIME」の誤答数の平均が 0.17 で、「AjaxIME」の誤答数の平均が 0.58 少なく有意（両側検定）な差を付けた（ $p < .05$ ）。ポルトガルのような学習経験がなく綴りになじみがない言語入力について「AjaxIME」は誤答を少なくさせることが明らかになった。

ヒンディー語において、3 方式の平均の分散分析の結果、誤答数では有意な差がみられなかった。

表 18 誤答数（対応のある t 検定（両側検定））

被験者	日本語		英語		ポルトガル語	
	誤答数A1	誤答数A2	誤答数B1	誤答数B2	誤答数D1	誤答数D2
H01	1	3	3	0	-	-
H02	1	1	1	1	-	-
H03	1	1	0	0	-	-
H04	1	1	1	0	-	-
H05	1	1	0	0	-	-
H06	0	0	0	0	-	-
P01	1	0	0	2	2	1
P02	3	0	0	0	1	0
P03	1	0	0	0	1	0
P04	0	0	2	1	3	0
P05	3	0	1	0	0	0
P06	1	0	0	0	1	0
平均	1.17	0.58	0.67	0.33	1.33	0.17
平均値の差 (t検定)	0.171		0.339		0.034	
5%水準*					*	
1%水準**						

表 19 誤答数（対応のある一元配置の分散分析）

被験者	ヒンディ語		
	誤答数C1	誤答数C2	誤答数C3
H01	1	2	4
H02	1	0	1
H03	4	2	4
H04	5	7	6
H05	2	3	3
H06	2	0	1
P01	-	-	-
P02	-	-	-
P03	-	-	-
P04	-	-	-
P05	-	-	-
P06	-	-	-
平均	2.50	2.33	3.17
一元配置の 分散分析	0.609		
5%水準*			
1%水準**			

(c) 回答率

「既存の入力方式」と「AjaxIME」の回答率(回答数/30)の平均を比較した(表 20、表 21)。

回答率の平均比較は、英語、ポルトガルでは有意な差はなかった。英語、ポルトガルのような直接入力方式の言語では回答率は「既存の入力方式」と「AjaxIME」との差はない。

日本語において、「既存の入力方式(A1)」と「AjaxIME(A2)」を比較した場合、「既存の入力方式」は回答率の平均が96.39%、「AjaxIME」は回答率の平均が90.56%であり、「AjaxIME」の回答率の平均が5.83%少なく有意(両側検定)な差を付けた($p < .05$)。これは、日本語入力は、普段から仮名漢字変換をベースとするInputMethodに慣れており、かつ入力の際の変換規則をへボン式に固定したため、被験者が普段使い慣れたInputMethodを用いた変換方式での高速な入力と比べて、初めて利用する「AjaxIME」では入力がしにくかったと考えられる。

ヒンディー語において、3方式の平均の分散分析の結果、有意な差がみられた($p < .01$)。「既存の入力方式(C1)」、「AjaxIME 詳細情報なし(C2)」、「AjaxIME 詳細情報あり(C3)」それぞれのペアにおける有意差を検出するために、回答率に対する多重比較(Tukey法)による分析を行った。

「既存の入力方式」と「AjaxIME 詳細情報なし」および、「既存の入力方式」と「AjaxIME 詳細情報あり」においては有意な差が見られた(共に $p < .05$)。しかしながら「AjaxIME 詳細情報なし」と「AjaxIME 詳細情報あり」においては有意な差は見られなかった。

従って、ヒンディー語において、「AjaxIME」が「既存の入力方式」よりも多くの文字入力ができることが明らかとなり、設定した仮説を確認できた。一方、詳細情報の有無が入力に与える影響については、充分確認することができなかった。

表 20 回答率（対応のある t 検定（両側検定））

被験者	日本語		英語		ポルトガル語	
	回答率A1	回答率A2	回答率B1	回答率B2	回答率D1	回答率D2
H01	100%	100%	100%	100%	-	-
H02	100%	100%	100%	100%	-	-
H03	100%	87%	100%	100%	-	-
H04	100%	100%	100%	100%	-	-
H05	100%	100%	100%	100%	-	-
H06	100%	90%	100%	100%	-	-
P01	100%	87%	100%	87%	40%	83%
P02	100%	100%	100%	100%	97%	100%
P03	100%	90%	100%	93%	90%	97%
P04	80%	83%	100%	100%	100%	100%
P05	100%	87%	100%	100%	67%	87%
P06	77%	63%	100%	87%	97%	100%
平均	96.39%	90.56%	100%	97.22%	81.67%	94.44%
平均値の差 (t検定)	0.013		0.096		0.117	
5%水準*	*					
1%水準**						

表 21 回答率（対応のある一元配置の分散分析）

被験者	ヒンディ語		
	回答率C1	回答率C2	回答率C3
H01	0.43	1.00	0.93
H02	0.40	1.00	1.00
H03	0.90	0.77	1.00
H04	0.73	1.00	1.00
H05	0.40	0.77	0.93
H06	0.40	0.93	1.00
P01	-	-	-
P02	-	-	-
P03	-	-	-
P04	-	-	-
P05	-	-	-
P06	-	-	-
平均	0.54	0.91	0.98
一元配置の 分散分析	0.007		
5%水準*	**		
1%水準**			

(2) ヒアリングによる評価

次に、言語毎の入力実験終了後、ユーザーに主観評価をヒアリングした結果を示す。

①日本語の AjaxIME での入力

プラス評価として、

- ・予測変換があるから良い
- ・同じ読みの異なり語を入力するときに Detailed Information があると良い
- ・外国語を入力するのには使えそう
- ・欧米圏の人には使いやすそう

マイナス評価として、

- ・ヘボン式は使い慣れない
- ・読み方がわからないときは難しい
- ・慣れていないため使い勝手に難がある

という意見がでた。

日本人にとって日本語入力は一番慣れたものであるため、既存の IM の入力速度があがるものの、AjaxIME の詳細情報については、評価が高かった。

②英語の AjaxIME での入力

- ・候補がでてくるのがよい
- ・長い綴りのときに有効
- ・日本語よりは既存方式との違いが無い感じがする

といったプラス評価がほとんどだった。要望として、候補としてリストアップされる文字列は key input string でのソート順ではなく、類似した単語が良いという意見があった。

③ヒンディー語の AjaxIME での入力

詳細情報無しより、詳細情報付きの評価が非常に高く、

- ・読み方ができるからやりやすい
- ・似た単語でも判断しやすい

というプラス評価を得た。

一方、

- ・ 転写規則を覚えるのがたいへん
 - ・ 転写規則の一部に違和感がある
- というマイナス意見もあった。

④ポルトガル語の AjaxIME での入力

- ・ 英語よりもさらに使いやすい。
 - ・ 言語によって違う可能性がある気がする
- というプラス評価を得た。

7. 6 まとめ

本研究により、情報資源の対象をインターネットで Web 公開されている Unicode (UTF-8) で記述されたあらゆるデータと置き、そのデータへアクセスするために必要な検索エンジンで指定する検索クエリーを入力するための多言語 InputMethod が実現できた。

本研究で実装したシステムは、各言語特有の言語解釈は行わず、入力文字列と出力文字列の対応を示す変換辞書のみによりキーボードから入力する ASCII 文字内の入力文字列を Unicode (UTF-8) で記述されたあらゆる言語に対応する出力文字列に変換して出力を行う方法をとる。従って、Unicode 規格化されたすべての言語への対応ができるといえる。また、実際にインド系文字 10 言語、アラビア文字 3 言語、日本語、英語、ポルトガル語の変換辞書を作成し、Web 上での入力が可能であることを確認済みであり、このようなユーザーの環境に依存しない多言語の InputMethod の実現は、これからの多言語化を支える基盤として、重要な試みといえる。

本研究では、どのような言語にも対応可能な InputMethod という特徴に加え、変換候補となる語彙に関する情報である詳細情報の表示機能を実装したところに新規性がある。詳細情報は、その言語に不慣れなユーザーが間違いなく所望の語彙入力ができ、かつ言語学習の補助としての機能となるという仮説の元に実装した機能である。実際に、ユーザーによる入力実験を行ったところ、ヒンディー語の実験では、既存の入力方式と本研究での提案方式を比較した場合、正答数と回答率におい

て本研究での提案方式のほうが良い結果となった。ただし、詳細情報がある場合とない場合では有意な差は見られなかったが、ヒアリングによるユーザー評価では詳細情報の存在が高く評価されている。

また、本研究では、インクリメンタルサーチによる文字変換が、入力しようとしている言語に不慣れなユーザーが正確な綴りに早く導かれるきっかけをあたえるものであるという仮説設定のもと、提案システムに実装している。ヒンディー語の実験では、既存の入力方式と詳細情報がない本研究での提案方式を比べた場合、正答数と回答率において本研究での提案方式の方が良い結果となっており、このことから詳細情報だけでなくインクリメンタルサーチによる入力が、入力しようとしている言語に不慣れなユーザーにとって有効であることが、明らかとなった。

さらに、本研究での提案方式は、既存の入力方式と比べヒンディー語やポルトガル語というユーザーの習熟レベルが低い言語の場合のほうが、正答数や誤答数、回答率において良い結果となっており、このことから、利用したい言語に不慣れな言語学習途上のユーザーが簡単に多言語入力を行える環境の提案を行えたと評価する。

今後の課題として、ユーザーによる入力実験でのヒアリングで指摘されたように、変換辞書については、ユーザーの転写規則に対する慣れが必要で、いままでのキーボード配列との違いなどが使用言語や個人の好みにより異なることから、多くのユーザーに利用してもらい、変換辞書のカスタマイズを行っていく予定である。

第八章 多言語 InputMethod の IVS 対応

8. 1 はじめに

本章では、多文字対応として Unicode 定義されている Ideographic Variation Sequence (IVS) を使用した異体字文字表示の実装を試みた。IVS は、Unicode Consortium で仕様定義されているが、現在まだフォント、入力、レンダリングエンジンが充分整備されていないため、一般には使用できない状態が続いている。有効性を確認するため、実装と実験を行い、今後の課題を整理する。

8. 2 IVS 対応に向けた環境の整理

IVS とは、「5. 3. 1 異体字」で示したように、多文字である漢字において、符号化されている漢字の異体字を表現可能とするコード列でしかない。つまり、実際に異体字をディスプレイ等に出力するためには、IVS が定義されているだけでは、まったく意味をなさず、IVS を入力する InputMethod、IVS 対応のフォントと IVS を解釈しフォントから適切なグリフを呼び出し表示するレンダリングエンジン、そしてそのレンダリングエンジンに対応するアプリケーション（テキストエディタやブラウザ）が必要である。

過渡期であるため、公式に IVS 対応している環境は 2009 年 12 月現在ほとんどない。

InputMethod については、現状で IVS 対応を表明している InputMethod はないため、「第七章 多言語 InputMethod の実装と評価実験」で実装した、多言語 InputMethod の IVS 対応を試みる。

フォントについては、Adobe Systems 社が IVD 登録時にも利用している小塚明朝 Pr6N や小塚ゴシック Pr6N、またグリフ wiki で公開している花園明朝 OT xProN⁶⁶が IVS 対応を表明している。今回は、IVS のフォント実装方法も確認対象とするため、オープンフォントである IPA フォントを使って IVS 実装を試みる。

⁶⁶ グリフ wiki 「花園明朝 OT とは」

http://glyphwiki.org/wiki/Group:mashabow_%E8%8A%B1%E5%9C%92%E6%98%8E%E6%9C%9D%E4%BB%AE

レンダリングエンジンとして、OS レベルでは Windows7 が対応しているが、表示できるアプリケーションはメモ帳だけである⁶⁷。これは、Windows7 で稼働するブラウザやオフィスソフトウェア等の多くのアプリケーションが独自のレンダリングをおこなっているため、必ずしも OS レベルの対応だけでは、IVS 実装の確認が出来ない状況を示している。

8. 3 IVS 対応の実装

多言語 InputMethod の IVS 対応および IPA フォントの IVS 対応を行い、Windows7 での異体字表示を行う。

8. 3. 1 IVS 対応フォントの作成

IPA フォントを利用して、IVS 対応フォントを作成した。

OpenType フォントの IVS 対応については、OpenType specification version 1.6(2009)の中で、cmap テーブル (Character To Glyph Index Mapping Table) の Format 14 (Unicode Variation Sequences) に仕様が明示されている⁶⁸。この仕様に従って、IPA フォントの改変を行った。IVS 追加の対象を、JIS X 0208:1997 から JIS X 0213:2004 で例示字形が変更された 168 字とした。

8. 3. 2 IVS 用変換辞書作成

多言語 InputMethod の IVS 対応においては、「7. 4. 3 変換辞書データ」の登録が必要となる。

変換辞書とは、key input string、output string、Detailed Information をタブ区切りしたテキストデータであるが、IVS 対応にあたっては、output string に異体字セレクタ (Ideographic Variation Selector Character : IVSC) を付加することになる。

例えば、葛飾区や葛城市の「葛」は、本来の字体を図 47 に示すような字体であ

⁶⁷ 2009 年 12 月現在、Microsoft 社は、Windows7 に対する IVS 対応は正式にはコメントしていない。技術者間による非公式情報に基づいて確認作業を行ったところ、メモ帳での異体字表示を確認した。また、Adobe 社は OS には依存せずに Acrobat9 で IVS 対応済みを公表しており、異体字表示切り替えが行えることを確認した。

⁶⁸ <http://www.microsoft.com/typography/otspec/cmap.htm>

ると主張しているが⁶⁹、多くのフォントでは、異なる字形が採用されているため、表示することはできない。

葛 飾 区 議 会 葛 城 市
か つ ら ぎ し

図 47 葛飾区や葛城市の例

そこで、多言語 InputMethod で葛飾区や葛城市の「葛」を所望の字形で表示するために、図 48 のような変換辞書テキストを作成した。図 48 は可視的に異体字セレクタを表示するために Mozilla Firefox3.55 上で表示した際の画面ダンプである。

```
katushikaku 葛飾区 名詞*  
katushikaku 葛0E0100飾区 名詞*  
katuragishi 葛城市 名詞*  
katuragishi 葛0E0100城市 名詞*  
a 逢0E0100ワ行五段* JIS2004字形  
a 逢0E0100ワ行五段* JIS97字形  
ame 飴 名詞* JIS2004字形  
ame 飴0E0100 名詞* JIS97字形  
ibara 茨 名詞* JIS2004字形  
ibara 茨0E0100 名詞* JIS97字形
```

図 48 IVS 用変換辞書

葛飾区の例では、1 行目は IVS を用いない通常の出力のために key input string : katushikaku、output string : 葛飾区、Detailed Information : 名詞とした。2 行

⁶⁹ 葛飾区議会「文字の表記について」：
<http://www.katsushika-kugikai.jp/10300.html>

目は、IVS を用いた異体字出力のために key input string : katushikaku、output string : 葛 U+0E0100 飾区、Detailed Information : 名詞とした。これにより、2 種類の葛のグリフを切り替えることができるか試すことができる。

8. 3. 3 MySQL への登録

IVS 用変換辞書を、MySQL に登録するにあたってテーブルの修正とそれに伴うプログラムの修正が必要となった。

今回実装に利用したリレーショナル DB である MySQL5.0.82 は、Unicode を扱うエンコード方式である UTF-8 をサポートしているが、UTF-8 の 4 バイト文字対応をしていないことがわかった。

UTF-8 は、図 49 のようなビットパターンになっており、3 バイトまでで 16bit、4 バイトまでで 21bit まで表現できる。3 バイトで表現できるのは Unicode の第 0 面である Basic Multilingual Plane 面 (U+0000~U+FFFF) であり、U+E0100 から U+E01EF の異体字セレクトは 4 バイト文字対応でなければ使用できない。

0xxxxxxx	(00-7f) 7bit
110yyyyx 10xxxxxx	(c0-df)(80-bf) 11bit
1110yyyy 10yxxxxx 10xxxxxx	(e0-ef)(80-bf)(80-bf) 16bit
11110yyy 10yyxxxx 10xxxxxx 10xxxxxx	(f0-f7)(80-bf)(80-bf)(80-bf) 21bit
111110yy 10yyyxxx 10xxxxxx 10xxxxxx 10xxxxxx	(f8-fb)(80-bf)(80-bf)(80-bf)(80-bf) 26bit
1111110y 10yyyyxx 10xxxxxx 10xxxxxx 10xxxxxx 10xxxxxx	(fc-fd)(80-bf)(80-bf)(80-bf)(80-bf)(80-bf) 31bit

図 49 UTF-8 のビットパターン

そこで、多言語 InputMethod 用に設計した DB における output string のデータ型が varchar であったところを、varbinary に変更し、さらに Ajax プログラム側で varbinary を変換するように改変を行った (図 50)。

結果として、図 51 のように IVSC を含む文字列を DB に登録することができた。ただし、表示は Web ブラウザから MySQL を操作するツールである phpMyAdmin の画面であるため、Web ブラウザが IVS 未対応であるため、異体字セレクト部分は

□で表示されている。

表示 構造 SQL 検索 挿入 エクスポート インポート 操作 空にする 削除

フィールド	種別	照合順序	属性	ヌル (NULL)	デフォルト値	その他	操作
<input type="checkbox"/> search_term	varchar(100)	utf8_general_ci		いいえ			<input type="checkbox"/>
<input type="checkbox"/> term	varbinary(255)			いいえ			<input type="checkbox"/>
<input type="checkbox"/> info	text	utf8_general_ci		いいえ			<input type="checkbox"/>

↑ すべてチェックする / すべてのチェックを外す チェックしたものを:

印刷用画面 リレーションビュー テーブル構造を確認する

1 個のフィールドを追加する テーブルの末尾 テーブルの先頭 指定カラムの後に

実行する

図 50 多言語 InputMethod の DB 構造

←T→	search_term	term	info
<input type="checkbox"/>	katushikaku	葛飾区	名詞*
<input type="checkbox"/>	katushikaku	葛 飾区	名詞*
<input type="checkbox"/>	katuragishi	葛城市	名詞*
<input type="checkbox"/>	katuragishi	葛 城市	名詞*
<input type="checkbox"/>	a	逢	ワ行五段* JIS2004字形
<input type="checkbox"/>	a	逢	ワ行五段* JIS97字形
<input type="checkbox"/>	ame	飴	名詞* JIS2004字形
<input type="checkbox"/>	ame	飴	名詞* JIS97字形
<input type="checkbox"/>	ibara	茨	名詞* JIS2004字形
<input type="checkbox"/>	ibara	茨	名詞* JIS97字形
<input type="checkbox"/>	inn	淫	名詞* JIS2004字形
<input type="checkbox"/>	inn	淫	名詞* JIS97字形
<input type="checkbox"/>	umaya	厩	名詞* JIS2004字形
<input type="checkbox"/>	umaya	厩	名詞* JIS97字形
<input type="checkbox"/>	esa	餌	名詞* JIS2004字形
<input type="checkbox"/>	esa	餌	名詞* JIS97字形
<input type="checkbox"/>	shaka	釈迦	名詞* JIS2004字形
<input type="checkbox"/>	shaka	釈迦	名詞* JIS97字形
<input type="checkbox"/>	ashi	芦	名詞* JIS2004字形

図 51 IVS 用変換辞書の DB への登録

8. 4 IVS 対応の検証

文字入力クライアント環境として、OS を Windows7、文字入力は Internet Explorer 8 上で多言語 InputMethod を用いて、IVS 対応の検証をおこなった。

図 52、図 53 は、葛城市を入力した例である。図 53 のように確定した文字の中の異体字セレクトは□で表示されており、異体字が表示されていない。これは、Internet Explorer 8 が IVS 対応できていないためである。

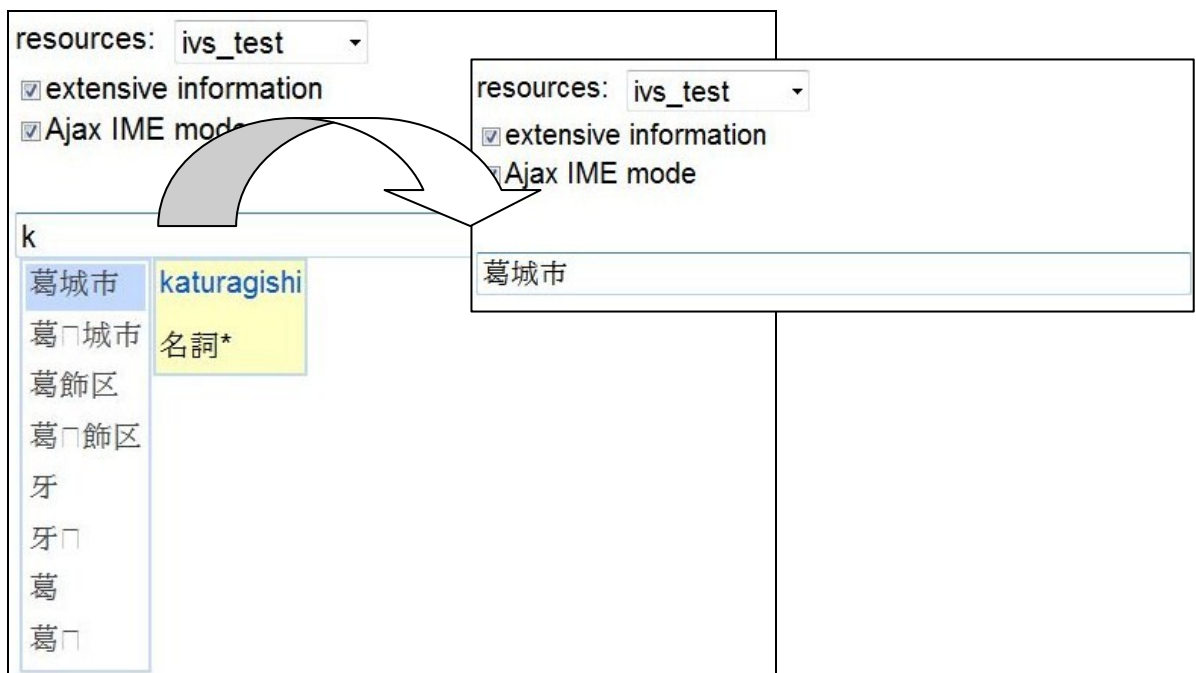


図 52 葛城市の入力 (IVSC なし)

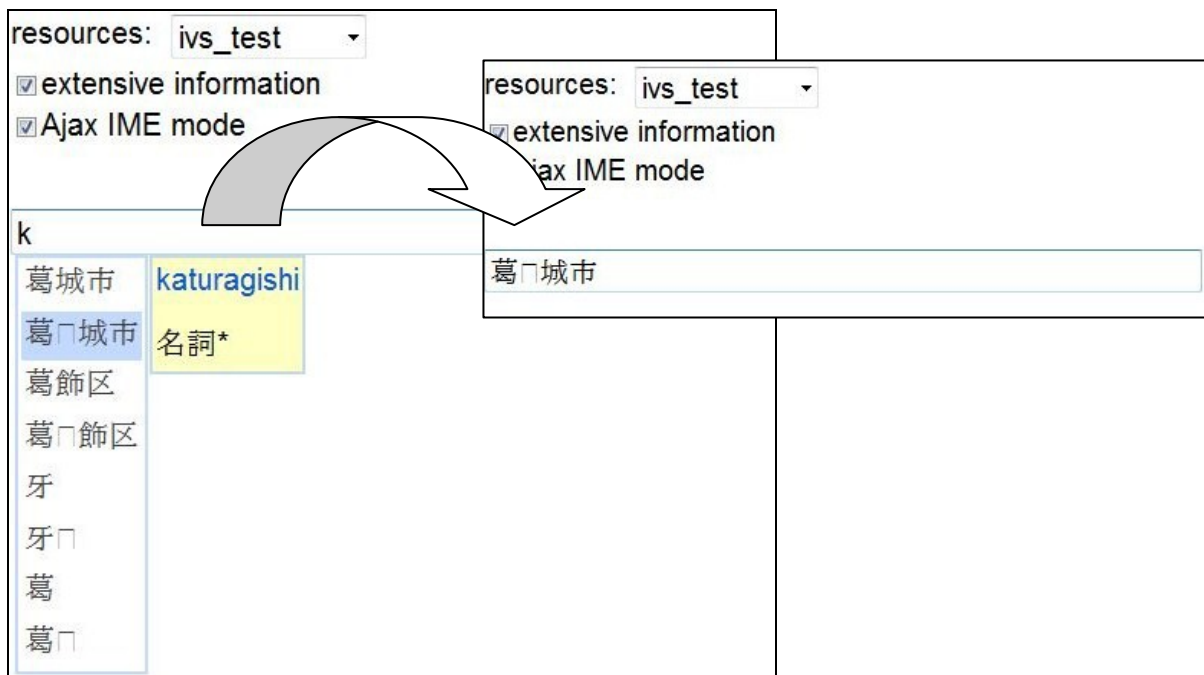


図 53 葛城市の入力 (IVSC あり)

そこで、Internet Explorer 8 で表示されている図 52、図 53 は、葛城市の文字列をメモ帳にコピーした。同様に「枢機卿」も多言語 InputMethod で入力しメモ帳にコピーしたものが、図 54 である。メモ帳上では、IVS を解釈し、異体字を表示していることが確認できた。

したがって、多言語 InputMethod により、IVS を文字コードとして間違いなく処理できていることが確認できた。

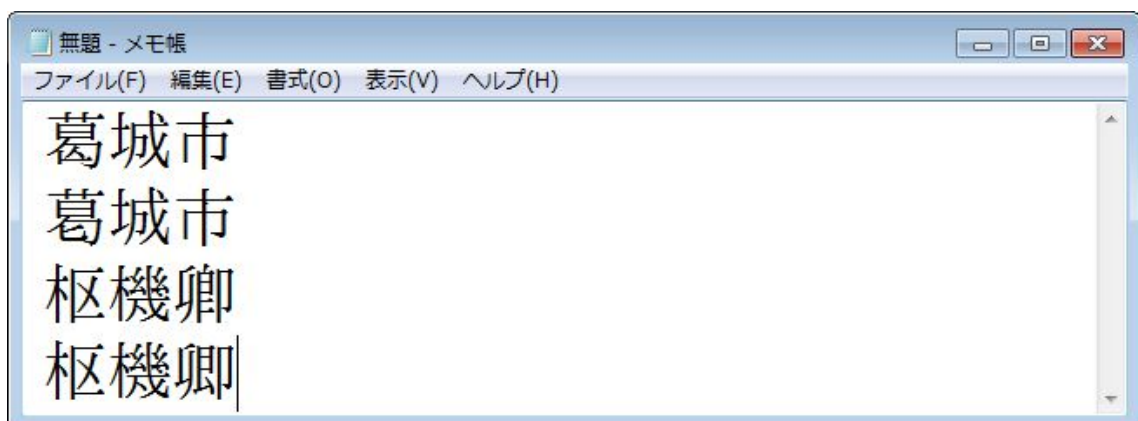


図 54 メモ帳での表示

8. 5 おわりに

本章で、IVS に対応したフォントおよび InputMethod を作成し、IVS を使用した異体字文字表示の実装を試みた。結果として、IVS 対応が過渡期であるため、表示環境に制約はあるものの、IVS により異体字表示をおこなうソリューションの展開は可能であることが確認できた。また、本研究で開発した AjaxIME のような IVS 対応 InputMethod および IVS 対応のパブリックフォントをオープンソースとして提供すれば、行政システム等の情報通信基盤における障害となっている異体字問題を解決するひとつの手段となりうることを同時に確認できた。

今後の課題としては、IVS のレンダリングを行うことができる OS 並びにアプリケーションを増やすことである。本章の検証のように、フォントおよび InputMethod が対応していても、表示側 (=レンダリング) が対応していないと、相互運用性が生まれない。

行政システムにおける文字処理環境の整備という視点において、オフィスソフト、Web ブラウザなどが IVS に対応することが期待される。

第三部 結論

第九章 まとめおよび今後の課題と展望

9.1 はじめに

本研究では、我が国の情報通信基盤としての文字処理環境について、日本語文字処理環境と多言語・多文字処理環境の2つの視点からの研究をおこなった。

本章では、それぞれで得られた知見をまとめ、さらに情報通信基盤として必要とされる文字処理環境の課題および展望について整理する。

9.2 まとめ

日本語文字処理環境の研究において、まず、第二章では、日本語文字処理環境の概括と問題の所在を明らかにした。この中で、日本語は表記に、表意文字である漢字、表音文字であるひらがな、カタカナ、さらにはラテン文字や記号等を使用する世界でも稀な言語であるため、日本語を文字にして ICT の中で利用するまでの過程として、印刷技術とコンピュータ化の歴史の中で日本独自の工夫を行ってきたことがわかった。

特に日本語を入力する InputMethod については、限られたキー配列の中でいかに簡単に、正確に文字を入力することができるかについて、さまざまな研究が行われ、レベルの高い入力が行える環境が整ってきた。

一方で、情報通信基盤を支える日本語文字処理環境においては、符号化文字集合として扱う文字数が多いという点で多くの課題を抱えていることがわかった。ICT における相互運用性、情報交換性を維持するためには、符号化文字集合の規格化が必須であり、国内・国際の文字規準が時代と共に整い、さらに変更されてきた。しかし、これまでに規格内の変更や規格間の不整合が存在しており、文字化けのような情報交換上の混乱を生じさせてきた経緯がある。また、符号化文字集合の変更は、文字を可視的にモニタや印刷物に映し出すためのフォントの変更を伴い、その為の作業の負荷や規格改定時の混乱によるユーザーへの負荷が強いられてきた。

IT 産業の歴史の中で、情報通信基盤となる日本語フォントは有償でサービスを提供する OS ベンダや SI ベンダが独自に調達することで、各 OS や SI ベンダの提供するクローズドな日本語環境での整合性をとってきた。そして、符号化文字集合の

規格が改変された際には、規格の不整合によるひずみを生み出してきた。一方で、インターネットが普及し、オープンソースソフトウェアにおける開発が推進される中、フォントについてもオープンであることが求められるようになってきた。

さらに、日本語の文字については、社会的、文化的側面での課題として国や地方自治体で扱う人名・地名などの外字問題がある。外字問題と呼ばれている問題は二つある。一つは JIS X 0208 や ISO/IEC 10646 などの情報交換用符号化文字集合で、包摂規準や Unification Rule 統合規則のために社会慣習上使い分けられている字形（＝異体字）が区別できないという問題である。もう一つは、そもそも符号化文字集合上に符号位置を持たない文字（＝外字）をどう取り扱うかという問題である。

本研究では、第二章で問題が明らかとなった情報通信基盤としてのオープンな日本語フォントの創出と、外字問題への対応について研究を行い、課題を明確にした。

第三章と第四章では、第二章で明らかとなった情報通信基盤としての日本語文字処理環境の問題の 1 つであるオープンな日本語フォントについての研究をおこなった。

第三章において、まず、情報通信基盤として必要とされてきた日本語パブリックフォントについて、技術と社会環境の変化により新たなドメインが必要とされてきた状況を整理し、日本語パブリックフォントという新たなドメインを提案した。

インターネットの普及により国際化が叫ばれる中、日本の IT 産業が競争力をもって開発を推進するため日本政府は OSS 普及推進を進めてきた。OSS における開発環境、異なる OS 間におけるマルチ OS 環境で、日本語の処理、表示を行うためには日本語フォントは必須である。しかしオープンな日本語フォントがなかったため OSS における日本語環境整備にあたって日本語フォントがボトルネックとなっていた。この問題を解決するために IPA が公開した IPA フォントを、日本語パブリックフォントという我が国の情報通信基盤として明確に位置づけた。日本語パブリックフォントが、情報交換性を保持するという制約と、相互運用性を高めるためのルール作りを行うための機能を持つことで、創発的な価値創造につながる新たなドメインとなりうることを確認した。

具体的な課題としては、フォントメンテナンス環境の整備として、オープンモジュールとしてのフォントの調整と規格の改定に伴うフォントのメンテナンスの必要性をあげた。相互運用性と技術イノベーション促進という視点では、日本語パブリ

ックフォントの広い利用が、異なる OS 間、異なるディストリビューション間の電子メディア交換において、同一のドキュメントが異なって見えてしまうという相互運用性問題を解消するとともに、多くの人々が日本語パブリックフォントを使うことで、さらに利用が促進され、より有効に利用されるというネットワーク外部性が働くことを示した。そして、オープンな情報交換の場がフォントやその関連技術に関する知識の共有を促進することで、新しい技術開発要素を生み出す、つまり技術イノベーションの促進が期待できることを示した。最後に環境作りとして、フォント開発環境の整備、パブリックフォント開発コミュニティの育成、パブリックフォントユーザーコミュニティの活性化を挙げ、オープンな環境でフォントおよび日本語の文字処理環境についての情報交換および技術育成の場を展開することを提案した。

第四章では、第三章で明らかにした情報通信基盤を支える日本語パブリックフォントのライセンスの研究を行った。

フォントは、その性格上プログラムとコンテンツの両方の要素を含んでいるが、加えてフォント特有の要素もあり、かつ日本語フォントには日本語特有の保護すべき要件として、日本語フォント特有の品質維持、間違っただ字形の氾濫阻止などが挙げられる。オープンなデジタル財の DRE としては、プログラムを対象とした GPL や、コンテンツを対象としたクリエイティブ・コモンズなどが確立しており、さまざまなデジタル財に用いられてきているが、日本語パブリックフォントに適したライセンスは存在していなかった。本研究では、相互運用性に富んだ情報通信基盤の確立へ寄与する日本語パブリックフォントの定義付けのもと、オリジナルフォントが保証する文字への信頼性確保の要件を明確にし、日本語フォントと情報通信基盤との関係性を示した。日本語パブリックフォントライセンスとしての要件を整理する中で、フォントにおいては、派生フォントの「フォント名」に対する使用許諾範囲が重要であることを明らかにした。また、情報通信基盤としての日本語フォントにおいてユーザーの混乱を招かないように、オリジナルフォントが保証する文字への信頼性を確保することが重要な要件であることを明らかにした。

本研究では、情報通信基盤となるべきパブリックフォントとして、日本語フォント特有の品質維持、間違っただ字形の氾濫阻止が行え、フォントの改変と再頒布が可能なパブリックフォント概念の創出と日本語フォントの健全な流通のために、日本語パブリックフォントライセンスである「IPA フォントライセンス v1.0」の具体的

な検討を行った。結果として「IPA フォントライセンス v1.0」は、国際的慣習に整合性の有るオープンソースライセンスとして OSI (Open Source Initiative) による OSD (The Open Source Definition) のオープンソース定義とを両立させ、DRE の視点に立脚したライセンスとして具体的要件を構築し日本で始めて OSI 承認を得て公開した。「IPA フォントライセンス v1.0」は、日本語フォントをデジタル財として自由に流通できる環境を整えることができるのみならず、国際的に利用可能なオープンなライセンスであり、日本語パブリックフォントと同様にフォントへの信頼性を確保したいと考える他言語のフォントにおいても充分適用可能なライセンスとして貢献することができる。

第五章では、情報通信基盤としての日本語の外字問題への1つの対応としてグリフデータベースの研究を行った。第三章、第四章で研究した日本語パブリックフォントは、相互運用性の高いフォントを提供することを重視して、フォントがオープンであることすなわち文字コードとしてもフォントフォーマットとしてもデファクトを採用することで、品質を保持するという考えのもと設計してきた。しかし、日本語パブリックフォントだけでは解決できない問題として外字問題が存在する。これまでの外字問題への対応としては、私用領域を用いたり、フォント名が異なるフォントファイルの恣意的な位置に字形も意味も全く異なる文字を割り付けたり、符号化せずに画像として表示したりといった対症療法的な方法をとっており、ネットワーク環境における情報資源としての活用を前提とした提案が行われていなかった。これでは、電子化文書としてのデータの互換性、検索性、相互運用性を著しく妨げることとなってしまう、かつ、我が国の情報通信基盤として各官公庁、地方自治体を含めた行政システムが独自システム開発を行うこととなりコストも膨大な額に上っている。

一方、漢字の扱いについては、情報交換性や相互運用性といった観点からは、文字の使用を情報交換用符号化文字集合の範囲内に止めることが望ましいが、地名や人名の表記に関しては、居住地への愛着心や父祖への尊敬心といった個人のアイデンティティに関わることでもあり、字形の使い分けへの要求を拒否することも困難である。したがって、異体字、外字を我が国の情報通信基盤としてどのように扱うべきかについての提案は重要である。

本研究では、異体字に関しては、ISO/IEC JTC1/SC2 および Unicode Consortium

により規格化された IVS、IVD を利用したアーキテクチャにより、フォントに異体字を実装することを検討した。フォントに IVS を利用して異体字を実装した場合、異体字を出力するためには、文字処理環境として、入力 (InputMethod)、処理 (各種アプリケーション)、出力 (フォント、レンダリング) のすべてが対応しなければ成立しないことを明らかにした。そして、フォント側の環境として、IVS を利用した異体字追加フォントを作成しやすい環境作りが必要であることを示した。

外字に関しては、オープンな環境での情報交換や相互運用性といった観点から、あくまでフォント化するのではなく文字画像として扱うことが望ましい。文字画像として扱うにあたっては、今後のネットワーク環境で自由度が高く、文字列の中に違和感なく埋めこむことの出来る汎用的なフォーマットであることが要件となる。研究の結果、SVG (Scalable Vector Graphics) Tiny 1.2 準拠が、インライン用画像フォーマットとして適していることを示した。

本研究では、異体字と外字を情報通信基盤として活用できる環境作りとして、オープンなネットワーク環境に文字画像を公開し、そこから異体字や外字を検索しダウンロードし、異体字は IVS を利用した異体字追加フォントを作成するために用い、外字は文字画像として利用できるようなグリフデータベースの提案と実装を行った。

グリフデータベースは、検索性が重要であることから、文字のメタデータの検討、検索機能の検討を行った。実装したシステムを運用評価したところ、漢字情報としては十分なメタ情報を保有したシステムであり、検索性に優れていることが確認できた。

第二部では、多言語・多文字処理環境の整備のための研究を行った。

まず、第六章では、多言語・多文字環境の概括と問題の所在を明らかにした。現在のインターネット環境における使用言語は、英語以外の言語が 60% を上回っているという統計結果でも明らかのように、多言語による情報の発信と蓄積が増加している。Unicode の整備によりマイナー言語への対応にまで情報交換用符号化文字集合の範囲が広がる中、多言語・多文字の文字処理環境については、まだ充分整備されているという状況にはない。

我が国における多言語・多文字処理環境の研究としては、入力、処理、出力に分けた場合、処理環境である機械翻訳分野の研究が長年進められていた。しかし、機械翻訳の品質は未だ十分ではなく、その前提で NICT 等が研究推進している「言語

グリッド」は、多言語サービス基盤としてインターネット上の言語資源(対訳辞書, シソーラス, コーパス) や言語処理機能(言語の解析、機械翻訳、言い換え)をセマンティック Web の技術により連携されることにより、世界規模の連携で言語の壁を越える試みである。各国からの言語資源を蓄積することで、各国の言語サービスが充実する事となるが、現在カバーしている言語は欧米各国の言語と、中国語、韓国語、タイ語などの比較的 IT 化が早くから進んでいてアジア地域の言語に限られている

多言語・多文字処理環境を、日本語やラテン文字以外を対象とした場合、文字を入力する環境を整備し、実際に文字入力をするには非常に難しい状況にあることがわかった。また出力環境もフォントやレンダリング実装が OS レベルに依っており、マイナー言語については、未だサポートされていない状況にある。さらに、国際化する中、国内外のインターネットカフェ等のオープンな環境で、Web 上の検索を行う際にも、検索するための文字が入力できない、という言語の障壁が立ち上がる。

本研究では、出力環境は今後の OS レベルの対応を待つこととして、まず、多言語の情報資源にアクセスするための文字入力方法への取り組みを行う。着眼点はこれまでの文字入力が、ローカルホストの OS レベルに依存しており、多言語文字入力を行う場合には、その言語用の入力方法のインストールやキーボード配列の選択が必要であり、また、多くの言語の場合は、キーボード配列に支配された入力を余儀無くされていたところを、ユーザー環境に縛られない入力方法とするところにある。

第七章では、実際に Unicode 符号化によりテキスト化されたあらゆる多言語情報資源へのアクセス方法として、ユーザー環境に縛られないインターネットブラウザでの入力操作による多言語 InputMethod を提案した。方法としては、OS や対象言語の制約を持たない文字入力手段として、ASCII 領域内の文字を入力文字として利用し、出力文字に各言語コードを対応させる変換辞書を利用した。この変換辞書へのアクセスには、インターネットブラウザで入力操作を行えるように Ajax (Asynchronous JavaScript + XML) 技術を用いたシステム実装を行った。さらに、利用したい言語に不慣れな言語学習途上のユーザーでも簡単に間違いなく所望の言語の情報資源にアクセスできるようなユーザーインターフェースとして、インクリ

メンタルサーチによる文字検索と変換候補となる語彙に関する情報を表示する詳細情報表示機能を加えた。システム実装を行い、ユーザーによる入力評価実験をおこなったところ、普段利用している日本語については、評価が低かったが、不慣れな言語（実験では、英語、ヒンディー語、ポルトガル語）においては、インクリメンタルサーチ及び詳細情報表示機能により、入力がやりやすくなったという評価を得た。マイナス評価としては、変換辞書の割り当てに慣れていない事に対する違和感があがったため、今後、変換辞書について検討する必要がある。

第八章では、第七章で構築した多言語 InputMethod を利用して、日本語の多文字に対応した入力環境の実証として、Unicode で規格化された Ideographic Variation Sequence (IVS) 対応による異体字表示の実装を行った。IVS に関する技術情報を元に、実際にフォントおよび多言語 InputMethod に異体字を実装し、IVS 解釈可能な環境を構築して、異体字の入力と表示が可能であることを検証した。

結果として、IVS 対応が過渡期であるため、表示環境に制約はあるものの、IVS により異体字表示をおこなうソリューションの展開は可能であることが確認できた。また、本研究で開発した多言語 InputMethod のような IVS 対応 InputMethod および IVS 対応のパブリックフォントをオープンソースとして提供すれば、行政システム等の情報通信基盤における障害となっている異体字問題を解決するひとつの手段となりうることを同時に確認できた。

この二部からなる研究を顧みると、日本語環境については、技術そのものというよりは、技術についての、情報交換性や相互運用性を考慮した制度設計、ソリューション展開についての研究となり、多言語・多文字処理環境については、まさに技術的にどのようにアクセスするかという手法の研究となった。一連の日本語文字処理環境の研究は、これまで相互運用性の視点で正面から手を付けてこなかった情報通信基盤整備の一助となると信じている。また、成熟しつつある高度な日本語文字処理技術を応用し、多言語・多文字処理環境についてもより研究をすすめることで、グローバル社会において日本だけでなく世界レベルで競争力を持つ我が国の情報通信基盤が整備され、さらには、海外への展開を行うことができることを願っている。

9. 3 今後の課題と展望

今後は、実践例としてあげた IPA フォントに対して、本研究で提案した日本語パ

ブリックフォントの課題を基にデザインした機能実装をおこない、その結果をフィードバックすることで、再度問題の抽出、デザインの変更をおこない、情報通信基盤として進化させていくことが課題である。そして、長期スパンでの日本語パブリックフォントおよび日本語パブリックフォントライセンスが及ぼした影響を観察、評価し、実社会と日本語パブリックフォントの関係を明らかにする。また、日本語パブリックフォントのメンテナンス活動をフォローし、より良い日本語環境の基盤となる情報発信、情報共有のあり方を検討提案し実践することで、日本語フォントのあるべき姿について提案するとともに、日本のソフトウェア産業、日本語を使う全ユーザーへのフィードバックを行うことがイノベーション促進へ寄与することが期待できる。図 55 にこれから目指そうとしている日本語フォントの位置づけを示す。

グリフデータベースについては、日本語パブリックフォントのサブセットとして、我が国の情報通信基盤の一助となるようオープンに公開することで、日本語文字処理環境の相互運用性の確保につながることを期待できる。課題としては、漢字の知識のないユーザーがグリフ検索を行うことのできる手書き文字検索や、類似文字の違いを表現するインターフェースなどの直感的なインターフェースの追加があげられる。また、オープンに公開する際には、グリフデータベースのメンテナンス・管理等の運用面での手法を確立することも課題である。

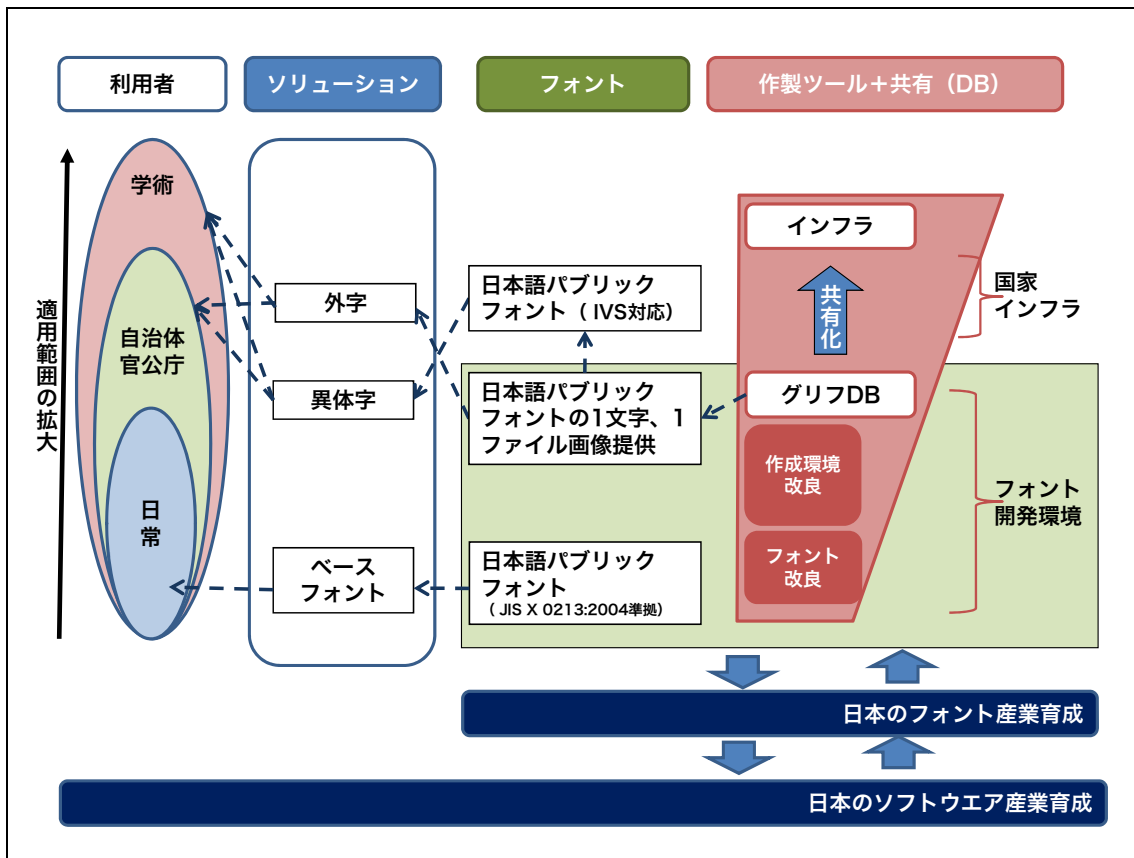


図 55 日本語パブリックフォントの位置付け

さらに、日本語パブリックフォントとグリフデータベースを利用して実装可能となる IVS を用いた異体字ソリューションでは、本研究成果をベースに自治体を含めた本格的な実証実験を行う事で、運用面等の課題の洗い出しを行う必要がある。そして、さらなる IVD 登録とフォントへの異体字収納が必要を行っていく。また、IVS のレンダリングを行うことができる OS 並びにアプリケーションを増やすこと相互運用性を高めていく。

本研究で定義した日本語パブリックフォントは、多くの Linux ディストリビュータと話し合いを重ねて、現在、Debian と Fedora の 2 大 Linux ディストリビューションで導入済みとなった。OpenOffice.org も標準バンドルを表明した。さらに相互運用性を高めたパブリックフォントとして確立させていくためには、Windows 環境、Mac OS X 環境への浸透を推進し、官公庁や学術分野における異体字ニーズも満たして行く必要がある。

多言語・多文字環境については、本研究ではターゲットユーザーを日本在住の日

本人を中心として、多言語 InputMethod の研究を行ったが、今後はアジア諸国などの発展途上国も対象としていきたい。日本政府は情報通信格差（デジタル・ディバイド）のための国際協力として、e-Japan 重点計画 2003（IT 戦略本部、2003）のなかで「IT を軸とした新たな国際関係の展開」を掲げている。また、今後予想される外国人労働者受け入れに際し、多言語化・他文化化が急速に進むことにともなう膨大なコスト（西垣、2001:18）を考慮すると、多言語・多文字環境への取り組みは重要な課題といえる。アジアの中の日本、多文字圏として多くのノウハウを持つ日本がデジタル・ディバイドを是正し、外国人受け入れ体制を強化するきっかけとして、多言語・多文字環境を積極的にリードしていくことが、他国の経済発展とともに、日本の経済発展にもつながることを期待する。

参考文献

- 青島矢一、武石彰（2001）「アーキテクチャという考え方」『ビジネス・アーキテクチャ』有斐閣、pp27-70
- 阿辻哲次（1994）『漢字の字源』講談社
- Bitstream, Inc.（2003）「Bitstream Vera Fonts Copyright」
http://www.gnome.org/fonts/#Final_Bitstream_Vera_Fonts
- 文化審議会国語分科会（2005）『国語分科会で今後取り組むべき課題について』
http://www.bunka.go.jp/kokugo_nihongo/bunkasingi/pdf/20050202.pdf
- Carliss Y. Baldwin , Kim B. Clark（2000）“Design Rules: The Power of Modularity” Mit Pr（=安藤晴彦訳（2004）『デザイン・ルール モジュール化パワー』東洋経済）
- CHO HEERYON、稲葉利江子、石田亨、高崎俊之、森由美子（2006）「絵文字コミュニケーションにおけるセマンティクス」『情報処理学会研究報告』 知能と複雑系 2006(110)、pp.1-8
- Creative Commons(2009)「About Licenses」<http://www.creativecommons.org/>
- 円満字二郎(2005)『人名用漢字の戦後史』岩波新書
- Eric, S Raymond（1999）、山形浩夫訳（2000）『伽藍とバザール(The Cathedral and the Bazaar)』<http://cruel.org/freeware/cathedra.html>
- Free Software Foundation, Inc.（2009）『GNU Licenses FontException』
<http://www.fsf.org/licensing/licenses/gpl-faq.html#FontException>
- 藤幡正樹（2009）『不完全な現実—デジタル・メディアの経験』NTT 出版
- 後藤晃（2001）「イノベーション・マネジメントとは」『イノベーション・マネジメント入門』一橋イノベーション研究センター編、日本経済新聞社
- 羽田亨（2010）「インフラストラクチャー」『日本大百科全書』小学館、
<http://100.yahoo.co.jp/detail/インフラストラクチャー/>
- 林紘一郎（2005）「第5章デジタル創作物と電子的権利制御」『クリエイティブコモンズ—デジタル時代の知的財産権』NTT 出版

- 林紘一郎 (2008) 『ネットワークにおけるデジタルコンテンツ取引流通フレームワークに関する調査研究』財団法人デジタルコンテンツ協会
- 林隆男 (1996) 『書体を創る』ジャストシステム
- Helmut Presser、饒田 収 訳 (1973) 『書物の本—西欧の書物と文化の歴史 書物の美学』法政大学出版局
- H.G.ウェルズ (1966) 『世界史概観』岩波書店
- 日立 (2007) 『戸籍統一文字』
http://www.hitachi.co.jp/Prod/comp/app/kosekionline/yougo_0482.html
- 堀修、三田雄志 (2001) 「テロップ認識のための映像からのロバストな文字部抽出法」『電子情報通信学会論文誌』 J84-D-II(8)、pp.1800-1808
- 法務省(2004) 『戸籍統一文字 使い方』
http://kosekimoji.moj.go.jp/kosekimojidb/jsp/mojidb/people_help.html#koseki
- 池田佳代、沼田秀穂、兼子正勝 (2009) 「高品質日本語オープンフォント活動の試みと社会的意義」『情報社会学会学会誌』 Vol.4 No.1,pp.105-119
- 生貝他 (2006) 「クリエイティブ・コモンズの深化と変容」情報管理 Vol. 49、No. 10、pp.576-585
- 稲葉利江子、山下直美、石田亨、葛岡英明 (2009) 「機械翻訳を用いた 3 言語間コミュニケーションの相互理解の分析(<特集>異文化コラボレーション論文)」『電子情報通信学会論文誌』 J92-D(6)、pp.747-757
- 井上史雄 (2001) 『日本語は生き残れるか—経済言語学の視点から』PHP 研究所
- 印刷博物館 (2009) 『百万塔陀羅尼』
http://www.printing-museum.org/collection/looking/02_01.html
- 石田亨 (2008) 「言語グリッドと異文化コラボレーション」『電子情報通信学会誌』 91(6)、pp.515-517
- 市村由美、齋藤佳美、木村和広、平川秀樹(2002) 「入力予測機能を組み込んだ仮名漢字変換システム」『電子情報通信学会論文誌』 J85-D-II(12)、pp.1853-1863
- 石崎俊 (2006) 『「汎用電子情報交換環境整備プログラム」4年間の成果』社団法人 情報処理学会 情報規格調査会、<http://www.itscj.ipsj.or.jp/jp/mojidb.html>
- 伊藤佳美 (2007) 「オープンソース・パラダイムシフト」『UNISYS TECHNOLOGY

- REVIEW』第 94 号、pp.3-15
- IT 戦略本部 (2003) 『e-Japan 重点計画 2003』
<http://www.kantei.go.jp/jp/singi/it2/kettei/030808honbun.pdf>
- James M. Utterback (1998) 『イノベーション・ダイナミクス—事例から学ぶ技術戦略』有斐閣
- Jef Raskin (2000) 『The humane interface: new directions for designing interactive systems』Addison-Wesley Publishing Co.
- 情報規格調査会 (2002) 『文字コード標準体系専門委員会報告書』社団法人情報処理学会
- 情報通信審議会答申 (2007) 『我が国の国際競争力を強化するための ICT 研究開発・標準化戦略』
http://www.soumu.go.jp/menu_news/s-news/2008/pdf/080627_6_bs1.pdf
- 上地宏一 (2002) 「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超える次世代漢字処理の提案」『漢字文献情報処理研究』No.3、pp. 4-13
- 上地宏一 (2008) 「漢字字形情報管理システムの構築と提案-日本漢文学研究への応用」『日本漢文学研究』第 3 号、pp.177-199
- 経済産業省 (2004) 『JIS 漢字コード表の改正について』
<http://www.meti.go.jp/kohosys/press/0004964/0/040220jis.pdf>
- Ken Lunde (1999) 『CJKV Information Processing』Oreilly & Associates Inc
- 木村忠正 (2001) 『デジタルデバインドとは何か』岩波書店
- 北川達朗 (2007) 「欧州の公的組織における OSS 化事情と日本市場における考察」『UNISYS TECHNOLOGY REVIEW』第 94 号、PP.17-35
- 清田公保、江崎修央、伊藤和之、伊藤和幸 (2007) 「中途視覚障害者の学習支援を目的としたペン入力学習ノート"Pen-Talker"の開発(福祉と知能・情動・認知障害, 福祉と音声処理, 一般)」『電子情報通信学会技術研究報告. TL, 思考と言語』106(485)、pp.25-30
- 小林章 (2005) 『欧文書体 その背景と使い方』美術出版社
- 小林龍生 (2005) 「標準化の政治社会学--UCS 標準化からのケーススタディ」『漢字文献情報処理研究』(6) 好文出版、pp.168~173
- 国語審議会 (2000) 『表外漢字字体表』

- http://www.mext.go.jp/b_menu/shingi/12/kokugo/toushin/001218.htm
- 国領二郎 (1999) 『オープン・アーキテクチャ戦略』ダイヤモンド社
- 国領二郎 (2003) 「ネットワーク上に構築される協働の場」『ネットワーク社会の知識経営』、NTT 出版、pp65-124
- (財)国際情報化協力センター (2005) 『アジア情報化レポート．オープンソースソフトウェア動向 2005』
- (財)国際情報化協力センター (2008a) 『アジア情報化レポート 2008 中国』、pp. 63-67
- (財)国際情報化協力センター (2008b) 『アジア情報化レポート 2008 韓国』
- (財)国際情報化協力センター (2008c) 『アジア情報化レポート 2008 タイ』
- (財)国際情報化協力センター (2008d) 『アジア情報化レポート 2008 インド』
- 小宮山博史 (2003) 「フォントの歴史と現在を知る」『フォントの鉄則』(株)毎日コミュニケーションズ、pp.13
- 小宮山博史 (2005) 「上海から明朝体書体がやってきた」『タイポグラフィの世界 書体編』大日本スクリーン製造
- 小宮山博史 (2008a) 「日本の活字書体はどのように作られてきたか」『文字は語る』ワークスコーポレーション
- 小宮山博史 (2008b) 「書体分類の意味と過去の分類」『基本日本語活字集成』誠文堂新光社
- 小宮山博史 (2009) 『日本語活字ものがたり』誠文堂新光社
- 近藤勝則 (2000) 「デジタル・デバイスをめぐる研究課題と対策」『郵政研究所月報』郵政研究所
- K. Sakamura (1987) 『Multi-language Character Sets Handling in TAD』TRON Project 1987, Springer-Verlag
- 工藤友彦、成田長人 (2005) 「ハイビジョン番組に適した字幕表示パラメータの検討--最適な文字サイズと文字数」『映像情報メディア学会誌』巻 59(11)、pp.1669 ~1672
- クリエイティブ・コモンズ・ジャパン事務局 (2006) 「クリエイティブ・コモンズの理念と実践」『情報管理』、Vol. 49、No. 7、pp.387-391
- 桑原洋 (2006) 「オープンソースソフトウェア(OSS)発展への期待」『情報処理』47

- 卷4号、情報処理学会、pp.418-420
- 丸山温道(2006)「フォント・タイプフェイスの保護」『月刊パテント』Vol.59 No.1、pp24-26
- 町田和彦(2001)『華麗なるインド系文字』白水社
- 町田和彦(2008)『ニューエクスプレス ヒンディー語』白水社
- 松尾賢一、上田勝彦、梅田三千雄(1997)「適応しきい値法を用いた情景画像からの看板文字列領域の抽出」『電子情報通信学会論文誌』J80-D-2(6)、pp.1617-1626
- Microsoft Corporation(2001)「OpenType Specification - Overview」
<http://www.microsoft.com/typography/otspec/otover.htm>
- Microsoft Corporation(2008)「OpenType specification - The OpenType Font File」
<http://www.microsoft.com/typography/otspec/otff.htm>
- 三上喜貴(2000)「言語間デジタル・ディバイド」『長岡技術科学大学言語・人文科学論集』14、pp.83-94
- 三上喜貴(2005)「世界の文字と文字符号(前編)」『情報処理』46(8)、pp.919-924
- 美馬正司(2001)「デジタル・ディバイドとその是正の必要性」
<http://www2u.biglobe.ne.jp/~machi-IT/026.pdf>
- Miniwatts Marketing Group(2009)『INTERNET WORLD USERS BY LANGUAGE Top 10 Languages』
<http://www.internetworldstats.com/stats7.htm>
- 宮部真衣、吉野孝、重野亜久里(2009)「外国人患者のための用例対訳を用いた多言語医療受付支援システムの構築(<特集>異文化コラボレーション論文)」『電子情報通信学会論文誌』J92-D(6)、pp.708-718
- 宮原 諄二(2005)『「白い光」のイノベーション』朝日新聞社
- 森健一(1972)「文字読取装置における画像処理」『テレビジョン学会誌』26(11)、pp.949-958
- 森健一(1979)「日本語情報処理」『テレビジョン学会誌』33(5)、pp.380-385
- 守岡知彦、師茂樹(2004)「文字素性に基づく文字処理」『情報処理学会研究報告』Vol.2004、No.58、pp. 53-60
- 諸橋轍次(1958)『大漢和辞典』大修館書店
- 村上敬亮(2004)「オープンソースを巡る著作権論議と知的財産政策への示唆」
<http://www.tokugikon.jp/gikonshi/232tokusyu2.pdf>、特 技 懇 NO.232、

pp.16-28

- 永原康史 (2008) 「文字を組む方法 第5回技術と方法 (2) 写真植字」『文字の手帖』モリサワ
- 中塚智子 (2008) 「タイプフェイスの保護のあり方に関する調査研究」知財研紀要
http://www.iip.or.jp/summary/pdf/detail07j/19_05.pdf
- 直井靖 (1999) 「表外漢字字体表試案の読み方試論」『漢字問題と文字コード』太田出版
- 名和小太郎 (2004) 『デジタル著作権』みすず書房
- NICT (2009) 『言語グリッド』<http://langrid.nict.go.jp/jp/>
- 日本規格協会 (2000) 『JIS X 0213:2000 7ビット及び8ビットの2バイト情報交換用符号化拡張漢字集合』
- 日本タイポグラフィ協会 A 「文字の知的財産権に関するQ & A」『日本タイポグラフィ協会』http://www.typo.or.jp/info/morals/q_a.html
- 日本タイポグラフィ協会 B 「タイプフェイスに関する倫理綱領」『日本タイポグラフィ協会』<http://www.typo.or.jp/act/morals/morall.html>
- 西垣通 (2001) 「第一章 英語教育と英語公用語化論」「インターネットで日本語はどうか」岩波新書
- 野田納嘉子 (1999) 『語学王トルコ語』三修社
- 野口祐子、堀岡力 (2007) 「デジタルネットワーク時代の著作権処理と課題」『電子情報通信学会誌』90(2)、pp.101-105
- 沼田秀穂、福田豊 (2009) 「ライセンス真正性から見たシステム信頼の一考察」『情報通信学会学会誌』27(1)、pp.25-38
- 小形克宏 (2003) 「文字コードと文字セット」『フォントの鉄則』(株)毎日コミュニケーション、pp.280-311
- 小形克宏 (2004) 「JIS X 0213の改正は、文字コードにどんな未来をもたらすか」『文字の海、ビットの舟』
<http://www.watch.impress.co.jp/internet/www/column/ogata/>
- 岡崎毅 (2004) 「国際情報通信政策」『社会情報学ハンドブック』東京大学出版会、pp.116-119
- 大塚勇一郎 (2010) 「社会資本」『日本大百科全書』小学館、

- <http://100.yahoo.co.jp/detail/社会資本/>
- Opensource.org (2006) 『The Open Source Definition』
- <http://opensource.org/docs/osd>
- 裁判所ウェブサイト A、平成15年(ワ)第2552号 著作権侵害に基づく差止等請求事件
- <http://www.courts.go.jp/hanrei/pdf/ADC04D23E1CE042949256F17003B0ACD.pdf>
- 裁判所ウェブサイト B、主文
- <http://www.courts.go.jp/hanrei/pdf/9D8C086D4E975B3749256AD400268140.pdf>
- 佐藤敬之輔 (1963) 『日本字のデザイン』 丸善
- 佐藤敬幸 (2000) 『ISO/IEC 10646 (符号化文字集合: UCS) とは?』
- <http://www.itscj.ipsj.or.jp/tutorials/tu48.html> 情報処理学会 情報規格調査会 NEWSLETTER No.48)
- 佐藤大介、志築文太郎、三浦元喜、田中二郎 (2006) 「Popie : フローメニューに基づく日本語入力手法」『情報処理学会論文誌』 Vol.47(7)、pp.2305-2316
- 佐藤敬之輔 (1963) 『日本字のデザイン』 丸善
- 澤和宏、若林哲史、鶴岡信治、木村文隆、三宅康二 (2001) 「こう配特徴ベクトルと変動吸収共分散行列による手書き漢字認識の高精度化」『電子情報通信学会論文誌』 D-II J84-D-II、pp.2387-2397
- 関口正裕 (2006) 『汎用電子情報交換環境整備プログラム』 情報処理学会 情報規格調査会、<http://www.itscj.ipsj.or.jp/topics/mojidb0706.html>
- SIL (2007) 『SIL Open Font License』
- http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=OFL
- 白川静 (1984) 『字統』 平凡社
- 白川静 (1987) 『字訓』 平凡社
- 白川静 (1996) 『字通』 平凡社
- 曾根原登 (2004) 「デジタル流通システムと著作権」 『著作権の法と経済学』、林紘一郎 監修、勁草書房、第2章
- Stallman, Richard (2003) 『フリーソフトウェアと自由な社会』 アスキー

- 須藤修 (2004)「電子政府」『社会情報学ハンドブック』東京大学出版会、pp.132-135
- 相山敬士 (2005:123)「コモンズのための著作権法の基礎理論」『クリエイティブ・コモンズ—デジタル時代の知的財産権』NTT 出版
- 砂田薫 (2007)「情報政策史の時代区分に関する提案—経済産業省と情報産業を中心に—」『日本社会情報学会学会誌』第 19 巻 1 号、pp.45-57
- 鈴木俊哉、佐藤大 (2006)「インド系言語レガシデータの符号化方式自動識別クメール語フォントにおける非標準レガシ符号の調査」『電子情報通信学会技術研究報告』オフィスインフォメーションシステム 106(195), pp.1-8
- 高林哲、小松弘幸、増井俊之 (2002)「Migemo: 日本語のインクリメンタル検索」『情報処理学会論文誌』Vol.43(12)、pp.3698-3705
- 高橋裕樹、金大祐、中嶋正之 (2005)「ハングルの構造特徴を用いた情景画像からのハングル文字列領域抽出」『電子情報通信学会論文誌』J88-D-II(9)、pp.1808-1816
- 田中哲朗、岩崎英哉、長橋賢児、和田英一 (1995)「部品合成による漢字スケルトンフォントの作成」『情報処理学会論文誌』Vol.36、No.9、pp. 2122-2131
- 田中久美子、犬塚祐介、武市正人 (2003)「少数キーを用いた日本語入力」『情報処理学会論文誌』Vol.44(2)、pp.433-442
- 種村嘉高、小谷章夫、山崎聖一、密山幸男、尾上孝雄 (2006)「視覚特性を考慮した文字の黒み推定に関する一検討(システムオンシリコン,RFID 技術及び一般)」『電子情報通信学会技術研究報告. SIS, スマートインフォメディアシステム』106(374)、pp.69-74
- 田代秀一 (2006)「オープンソースソフトウェア・センターの設立(オープンソース事情)」『情報処理』情報処理学会、Vol.47、No.5(20060515)、pp. 540-542
- The Unicode Consortium (2006)『The Unicode Standard, Version 5.0』
- 特許庁総務部総務課制度改正審議室 (2006)『産業財産権法の解説』発明協会
- 東京外国語大学アジア・アフリカ言語文化研究所 (2005)『図説 アジア文字入門』河出書房新社
- 氏原基余司(2006)「常用漢字表の成立に至る漢字政策」『日本語学』明治書院 25(11)、pp.35~53
- 鷺巣敏行 (2009)「文字のユニバーサルデザイン--ユニバーサルデザイン視点から

見た読みやすい文字の研究開発経緯 (特集 印刷とユニバーサルデザイン)」『日本印刷学会誌』46(3)、pp.131～136

山口仲美 (2006) 『日本語の歴史』岩波新書

山口雄志、小谷賢太郎、堀井健 (2007) 「スクリーンボタンを応用した視線移動による文字入力方法の評価(コミュニケーション支援及び一般)」『電子情報通信学会技術研究報告・HCS, ヒューマンコミュニケーション基礎』107(59)、pp.47-52

安岡孝一、安岡素子 (1999) 『文字コードの世界』東京電気大学出版局

横山詔一、ロング エリク、米田純子、和田志子、黒田信二郎、下川和男 (2004) 「日本語 Web IME の開発と図書館情報検索システムへの実装」『情報処理学会研究報告, デジタル・ドキュメント』、pp.43-47

謝辞

本研究にあたり、お世話になりました多くの方々に感謝いたします。

主任指導教員である兼子正勝教授には、本研究の論理展開の形成から文章の細部に至るまで、たいへん細やかなご指導をいただきました。社会人である私へお気遣い頂き、お忙しい中での迅速なご対応に深く感謝いたします。

学位審査委員をご担当いただきました福田豊教授、吉浦裕教授、太田敏澄教授、高橋裕樹准教授、坂本真樹准教授には、広範なるご指導とご助言に、深く感謝申し上げます。

私が所属しておりますエクセリードテクノロジーおよび情報処理振興機構で本研究に関連する研究・開発を共に進めてきた沼田秀穂博士には、本研究に当たって様々な協力と助言を頂き、深く感謝申し上げます。

日本語オープンフォントに関する研究では、情報処理振興機構オープンソフトウェア・センターの田代秀一センター長に情報のご提供および貴重なコメントをいただき深く感謝申し上げます。

多言語 InputMethod の研究では、東京外国語大学アジア・アフリカ言語文化研究所の町田和彦教授に評価実験用データの提供および貴重なコメントをいただき深く感謝申し上げます。また三省堂殿より評価実験用データとして貴重な辞書データの提供をいただき深く感謝申し上げます。

実験に快く参加いただいた兼子研究室の皆様、東京外国語大学でヒンディー語を学ぶ皆様、そして友人達に深く感謝申し上げます。

最後に、社会人になって再び始めました学生生活と本研究を暖かく見守り励ましてくれた家族と友人に感謝申し上げます。本論文の完成と博士学位取得を楽しみにしていた父池田守男は、思いがけず 2009 年 12 月 25 日に他界してしまいました。父は、長年勤めた企業を定年退職後、かつての研究成果をまとめ、2002 年に博士(工学)学位を取得しました。その努力と目標を持って挑む姿勢は私に少なからず影響を与えています。本論文を亡き父に捧げるとともに、常に父と私を多方面で献身的に支えてくれた母池田愛子に深く感謝したいと思います。

研究業績

1. 関連論文の印刷公表の方法及び時期

(1) 論文題目：「情報通信基盤としての高品質日本語オープンフォントに関する検討」

著者：池田佳代、沼田秀穂、兼子正勝

印刷公表の方法：『情報社会学会学会誌,Vol.4,No.1』 pp.105-119

時期：平成 21 年 6 月

本論文との関連：「第二章日本語文字処理環境の概括と問題の所在」、「第三章日本語パブリックフォントの提案」で使用

(2) 論文題目：「情報通信基盤としての日本語パブリックフォントライセンスの研究：DRE の視点から」

著者：池田佳代、沼田秀穂、兼子正勝

印刷公表の方法：『情報通信学会学会誌 91 号 Vol.27 No.2』 PP.37-54

時期：平成 21 年 9 月

本論文との関連：「第二章日本語文字処理環境の概括と問題の所在」、「第四章日本語パブリックフォントライセンス」で使用

(3) 論文題目：「Proposal for a Multilanguage Text Input Support System that is Easy for Beginner Language Learners」

著者：Kayo IKEDA, Hideho NUMATA, Masakatsu KANEKO, Kazuhiko MACHIDA

印刷公表の方法： "3rd International Universal Communication Symposium (IUCS 2009)", Tokyo, Japan, pp.109-114

時期：2009, December 4

本論文との関連：「第六章多言語文字処理環境の現状」、「第七章多言語 InputMethod の実装と評価実験」で使用

2. その他（研究発表：海外）

Hideho NUMATA, Kayo IKEDA, Takahiko KAMAE, Noboru SONEHARA(2007/6), Experimental Investigation of Differences among Web Search Engine, The First Workshop on Information Credibility on the Web (WICOW), in conjunction with The 21st Annual Conference of the Japanese Society for Artificial Intelligence (JSAI 2007 in Miyazaki),pp.81-88

Manabu ITO, Mayumi KOIKE, Kayo IKEDA, Soichiro HIDAKA and Terumasa AOKI (2006/1), "KANSEI PERSPECTIVE": IMAGE CHARACTERISTICS EXPRESSION BY MANUAL DESCRIPTION FOR IMAGE RETRIEVAL, International Workshop on Advanced Image Technology (IWAIT in OKINAWA), pp. 291-296

IKEDA Kayo, NUMATA Hideho, OYAMA Yoshihito, AOKI Terumasa(2006/1), A New Distributed Behavioral Model for Non Time-based CG System, International Workshop on Advanced Image Technology (IWAIT in OKINAWA),pp.73-74

Hideho NUMATA, Kayo IKEDA(2002), Verification of the Khmer script implementation based on ISO/IEC 10646-1:2000, Asian Forum for Information Technology (AFIT in Lao PDR)

3. その他（研究発表：国内）

池田佳代、沼田秀穂、兼子正勝「情報通信基盤としての高品質日本語オープンフォントに関する検討」『情報社会学会 2009 年度年次研究発表大会』、於：JICA 地球ひろば講堂、pp.105-119

沼田秀穂、池田佳代、渡辺克也、曾根原登(2009/11)「サイバー犯罪と実社会との関連性研究」『情報処理学会 情報セキュリティ心理学とトラスト(SPT)研究発表会』、情報処理振興機構

沼田秀穂、池田佳代、曾根原登（2007/3）「主要検索エンジンサイトランキング調査とメタサーチエンジンの研究開発」『信頼の情報通信メカニズム ワークショップ国立情報学研究所情報信頼性研究会』、学術情報センター

金子利佳、沼田秀穂、池田佳代、釜江尚彦、曾根原登（2007/2）『デジタル文章流

通における伏せ字付文章の認知特性』電子情報通信学会「ヒューマン情報処理研究会」「パターン認識・メディア理解研究会」合同研究会，東工大，pp. 35-40

沼田秀穂、池田佳代、金子利佳、釜江尚彦、曾根原登（2006/6）「テキスト半開示方法の提案」『画像電子学会 第 34 回年次大会』，東邦大学習志野，pp.105-106

池田佳代、沼田秀穂、釜江尚彦、曾根原登（2006/6）『テキストインデキシング方法の性能評価』画像電子学会 第 34 回年次大会，東邦大学習志野，pp.103-104

沼田秀穂、池田佳代、釜江尚彦、曾根原登（2006/6）「テキストインデキシング方法の提案」『画像電子学会 第 34 回年次大会』，東邦大学習志野，pp.101-102

大山義仁、沼田秀穂、池田佳代、江村恒一、青木輝勝（2006/1）『BOID アルゴリズムにおけるクラスタ度評価』電子情報通信学会「人工知能と知識処理」，情報処理学会「知能と複雑系」合同研究会，ヒルトン小樽(北海道)，pp. 63-69

池田佳代、宮原諄二（2005/11）「デジタルコミュニケーション技術のイノベーション事例の考察」『経営情報学会 2005 年秋季全国研究発表大会』，中村学園（福岡）

沼田秀穂、池田佳代（2005/9）「デジタルメディアイノベーションと社会への影響」『日本社会情報学会第 10 回研究大会（社会情報学フェア 2005）』，京都大学，pp.40-41

沼田秀穂、池田佳代（2005/6）「IT によるコミュニケーション手法の変化と知識創造への影響」『経営情報学会 2005 年春季全国研究発表大会 D1-2』，早稲田大学，pp.22-25

池田佳代、沼田秀穂、青木輝勝（2005/3）『立体視における適切なテキスト表現と表現手法の拡張に関する研究』情報処理学会 オーディオビジュアル複合情報処理研究発表会、Vol. 2005, No. 23、沖縄（KDDI）、pp.79-84

池田佳代、沼田秀穂、青木輝勝（2004/10）『立体映像におけるクロマキー合成の効果に関する研究』画像電子学会第 213 回研究会、山形大学、Vol. 04-04-07，pp.43-47

沼田秀穂、池田佳代、青木輝勝（2004/10）『立体視における適切なテキスト表現と表現手法の拡張に関する研究』画像電子学会第 213 回研究会，山形大学，Vol. 04-04-08，pp.49-52

池田佳代、沼田秀穂、青木輝勝（2004/9）『立体映像におけるクロマキー合成の効

果に関する研究』FIT2004 第3回情報科学技術フォーラム、同志社大学
沼田秀穂、池田佳代、青木輝勝（2004/9）『立体視における適切なテキスト表現と
表現手法の拡張に関する研究』FIT2004 第3回情報科学技術フォーラム、同志
社大学

4. 表彰

池田佳代、沼田秀穂、兼子正勝「情報通信基盤としての高品質日本語オープンフォ
ントに関する検討」『情報社会学会 2009 年度年次研究発表大会』、於：JICA 地
球ひろば講堂、pp.105-119、プレゼンテーション賞

著者略歴

1. 学歴

- 平成 3 年 3 月末日 国立千葉大学工学部画像工学科卒業
- 平成 16 年 4 月 1 日 東京理科大学専門職大学院総合科学技術経営研究科総合科学技術経営専攻入学
- 平成 18 年 3 月末日 東京理科大学専門職大学院総合科学技術経営研究科総合科学技術経営専攻修了
- 平成 19 年 10 月 1 日 電気通信大学大学院電気通信学研究科人間コミュニケーション学専攻 博士後期課程入学

2. 職歴

- 平成 3 年 4 月 1 日 大日本スクリーン製造株式会社入社
- 平成 14 年 4 月末日 同社退職
- 平成 14 年 4 月 1 日 有限会社エクセリードテクノロジー設立取締役就任。現在に至る

以降、東京大学先端科学技術研究センター協力研究員、東京大学国際・産学共同研究センター科学技術振興特任研究員、国立情報学研究所共同研究員、東北大学電気通信研究所共同研究員を経て、現在、情報処理推進機構(IPA) オープンソフトウェア・センター研究員。

3. 所属学会

情報通信学会、情報処理学会、映像情報メディア学会、日本社会情報学会 (JASI)

付録

付録 1 多言語文字入力支援システムの評価実験解答用紙および配布資料

アンケート

① お名前 _____ ② 年齢 _____ 才 ③ 性別 男・女

④ 国籍 _____ ⑤ E-Mail _____

⑥ 使える言語、または勉強した事がある言語とそのレベルをお答えください。レベルについては、1 から 5 のどれかに○をつけてください。

言語	レベル
_____	1. 自由に使える 2. 読み書きができる 3. 読むことができる 4. なんとか読むことができる 5. 勉強したことがあるだけ
_____	1. 自由に使える 2. 読み書きができる 3. 読むことができる 4. なんとか読むことができる 5. 勉強したことがあるだけ
_____	1. 自由に使える 2. 読み書きができる 3. 読むことができる 4. なんとか読むことができる 5. 勉強したことがあるだけ
_____	1. 自由に使える 2. 読み書きができる 3. 読むことができる 4. なんとか読むことができる 5. 勉強したことがあるだけ
_____	1. 自由に使える 2. 読み書きができる 3. 読むことができる 4. なんとか読むことができる 5. 勉強したことがあるだけ

⑦ パソコンの使用経験 _____ 年

⑧よく使う OS にチェック (レ) を入れてください (複数回答可)。

WindowsOS Macintosh Linux その他 ()

⑨パソコンで日本語を入力する際に、入力ソフト (InputMethod) は何を使っているか、チェック (レ) を入れてください (複数回答可)。

ATOK MS-IME ことえり その他 () 知らない

パソコンで使用している日本語と英語以外の言語があれば、その言語を記入してください。その言語のパソコンでの使用状況にチェック (レ) を入れ、使用ソフトを書いてください (複数回答可)。

言語	使用状況
_____	<input type="checkbox"/> Web で情報を見る (使用ブラウザ _____) <input type="checkbox"/> メールやワープロソフトで文章を書く。 (使用ソフト _____) <input type="checkbox"/> その他 ()
_____	<input type="checkbox"/> Web で情報を見る (使用ブラウザ _____) <input type="checkbox"/> メールやワープロソフトで文章を書く。 (使用ソフト _____) <input type="checkbox"/> その他 ()
_____	<input type="checkbox"/> Web で情報を見る (使用ブラウザ _____) <input type="checkbox"/> メールやワープロソフトで文章を書く。 (使用ソフト _____) <input type="checkbox"/> その他 ()

以上です。

A-1 日本語

出題した日本語の検索ワードを既存の入力ソフトを使って入力し、Google で検索し、その検索結果を記入してください。

注) 出題した日本語については、漢字や送り仮名、カタカナ、ひらがなを正確に入力してください。

【練習問題】

検索ワード	読み	検索結果(件数)
堪兼ねる	たえかねる	
在銘	ざいめい	
タール癌	タールがん	
重罪	じゅうざい	
辺り	わたり	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	読み	検索結果(件数)
1	姿勢	しせい	
2	志向	しこう	
3	訴訟救助	そしょうきゅうじょ	
4	薄美濃	うすみの	
5	鉄砲狭間	てっぽうざま	
6	指向性	しこうせい	
7	すれば		
8	邑	ゆう	
9	タッチ		
1 0	二の舞	にのまい	
1 1	耳垢	みみあか	
1 2	計数管理	けいすうかんり	
1 3	塩井雨江	しおいうこう	
1 4	ユーモレスク		
1 5	来迎会	らいごうえ	
1 6	消防設備士	しょうぼうせつびし	
1 7	致傷	ちしょう	
1 8	山主	やまぬし	
1 9	出炉	でろ	
2 0	割り粥	わりがゆ	
2 1	ツーリストビューロー		
2 2	野郎	やろう	
2 3	空読み	そらよみ	
2 4	幼者	ようしゃ	
2 5	収納	すのう	
2 6	四つ門	よつもん	
2 7	炊ける	たける	

28	体育の日	たいいくのひ	
29	鳥肌	とりはだ	
30	説破	せっぱ	

A-2 日本語

出題した日本語の検索ワードを AjaxIME を使って入力し、Google で検索し、その検索結果を記入してください。

注) 出題した日本語については、漢字や送り仮名、カタカナ、ひらがなを正確に入力してください。

AjaxIME での入力は読みをローマ字で記述する方法です。

日本語のローマ字表記法はヘボン式を採用しています。ヘボン式については【資料 2】をご覧ください。

【練習問題】

検索ワード	読み	AjaxIME 入力	検索件数
堪兼ねる	たえかねる	taekaneru	
在銘	ざいめい	zaimei	
タール癌	タールがん	ta-rugan	
重罪	じゅうざい	juuzai	
辺り	わたり	watari	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	読み	AjaxIME 入力	検索件数
1	姿勢	しせい	shisei	
2	志向	しこう	shikou	
3	訴訟救助	そしょうきゅうじょ	soshoukyuujo	
4	薄美濃	うすみの	usumino	
5	鉄砲狭間	てっぽうざま	teppouzama	
6	指向性	しこうせい	shikousei	
7	すれば		sureba	
8	邑	ゆう	yu	
9	タッチ		tatti	
10	二の舞	にのまい	ninomai	
11	耳垢	みみあか	mimiaka	
12	計数管理	けいすうかんり	keisuukanri	
13	塩井雨江	しおいうこう	shioiukou	
14	ユーモレスク		yu-moresuku	
15	来迎会	らいごうえ	raigoue	
16	消防設備士	しょうぼうせつびし	shouboushi	
17	致傷	ちしょう	chishou	
18	山主	やまぬし	yamanushi	
19	出炉	でろ	dero	
20	割り粥	わりがゆ	warigayu	
21	ツーリストビュー ロー		tsu-risutobyu-ro-	
22	野郎	やろう	yarou	
23	空読み	そらよみ	sorayomi	
24	幼者	ようしゃ	yousha	
25	収納	すのう	sunou	
26	四つ門	よつもん	yotsumon	

27	炊ける	たける	takeru	
28	体育の日	たいいくのひ	taiikunohi	
29	鳥肌	とりはだ	torihada	
30	説破	せっぱ	seppa	

B-1 英語

出題した英語の検索ワードを既存の入力ソフトを使って入力し、Google で検索し、その検索結果を記入してください。

注) 出題した英語については、つづりを正確に入力してください。

【練習問題】

検索ワード	検索結果(件数)
public utility charges	
yield	
preconception	
orange juice	
long sleeves	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	検索結果(件数)
1	hot	
2	lifestyle	
3	ability	
4	spring	
5	on foot	
6	transparency	
7	tsunami	
8	enter	
9	onion	
1 0	executive	
1 1	young	
1 2	the opposite side	
1 3	truffle	
1 4	pioneer	
1 5	manifestation	
1 6	public	
1 7	receive	
1 8	manage	
1 9	margin	
2 0	soon	
2 1	treat	
2 2	commandment	
2 3	appearance	
2 4	feces	
2 5	the winter solstice	
2 6	system	
2 7	window	

2 8	gills	
2 9	nutmeg	
3 0	help	

B-2 英語

出題した英語の検索ワードを AjaxIME を使って入力し、Google で検索し、その検索結果を記入してください。

注) スペースを入力したい場合だけは、アンダーバー “_” を入力してください。

【練習問題】

検索ワード	検索結果(件数)
public utility charges	
yield	
preconception	
orange juice	
long sleeves	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	検索結果(件数)
1	hot	
2	lifestyle	
3	ability	
4	spring	
5	on foot	
6	transparency	
7	tsunami	
8	enter	
9	onion	
1 0	executive	
1 1	young	
1 2	the opposite side	
1 3	truffle	
1 4	pioneer	
1 5	manifestation	
1 6	public	
1 7	receive	
1 8	manage	
1 9	margin	
2 0	soon	
2 1	treat	
2 2	commandment	
2 3	appearance	
2 4	feces	
2 5	the winter solstice	
2 6	system	
2 7	window	

2 8	gills	
2 9	nutmeg	
3 0	help	

C-1 ヒンディー語

出題したヒンディー語の検索ワードをソフトキーボードで入力してください。

ソフトキーボードでは、母音記号は、子音字の後で入力します。

入力したキーワードは Google の検索フィールドにコピー&ペーストして検索し、その検索結果のヒット件数を記入してください。

注) 出題したヒンディー語については、つづりを正確に入力してください。

【資料1】 Google 検索の方法、の「3. Soft Keyboard を用いた入力」のように、検索を行い、結果を記録します。

キー配列については、【資料3】 ソフトキーボードのキー配列、をご確認ください。

【練習問題】

検索ワード	検索結果(件数)
बिच्छू	
धोना	
फटना	
चूतड़	
गुड्डा	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	検索結果(件数)
1	कहानी	
2	पुदीना	
3	खीरा	
4	प्रणाली	
5	अंगूठी	
6	गाड़ी	
7	नौसेना	
8	कुर्सी	
9	निराला	
1 0	पिछला	
1 1	निमंत्रण	
1 2	नाम	
1 3	पोस्टर	
1 4	बैंक का एकाउंटेंट	
1 5	चलना	
1 6	बाज़ीगर	
1 7	पीलिया	
1 8	पंडा	
1 9	चरखा	
2 0	किरण	
2 1	देर	
2 2	उछालना	
2 3	बाप	
2 4	चौड़ाई	
2 5	पोता	
2 6	छल्ला	
2 7	पक्षपात	

2 8	क्रिया	
2 9	ओला	
3 0	खूबसूरत	

C-2 ヒンディー語

出題したヒンディー語の検索ワードを AjaxIME を使って入力し、Google で検索し、その検索結果を記入してください。

注) スペースを入力したい場合だけは、アンダーバー “_” を入力してください。

【資料1】 Google 検索の方法、の「2. AjaxIME からの Google 検索」のように、検索を行い、結果を記録します。

ヒンディー語の転写規則は、【資料2】をご覧ください。

【練習問題】

検索ワード	AjaxIME 入力	検索結果(件数)
बिच्छू	bicc'uu	
धोना	d'onaa	
फटना	p'aTanaa	
चूतड़	cuutaRa	
गुड्डा	guDDaa	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	AjaxIME 入力	検索結果(件数)
1	कहानी	kahaanii	
2	पुदीना	pudiinaa	
3	खीरा	k'iiraa	
4	प्रणाली	praNaalii	
5	अंगूठी	AguuT'ii	
6	गाड़ी	gaaRii	
7	नौसेना	nausena	
8	कुर्सी	kursii	
9	निराला	niraalaa	
1 0	पिछला	pic'alaa	
1 1	निमंत्रण	nimaMtraNa	
1 2	नाम	naama	
1 3	पोस्टर	posTara	
1 4	बैंक का एकाउंटेंट	baiMka_kaa_ekaauMTeMTa	
1 5	चलना	calanaa	
1 6	बाज़ीगर	baaziigara	
1 7	पीलिया	piiliyaa	
1 8	पंडा	paMDaa	
1 9	चरखा	carak'aa	
2 0	किरण	kiraNa	
2 1	देर	dera	
2 2	उछालना	uc'aalanaa	
2 3	बाप	baapa	
2 4	चौड़ाई	cauRaaii	
2 5	पोता	potaa	
2 6	छल्ला	c'allaa	

2 7	पक्षपात	pak\$apaata	
2 8	क्रिया	kriyaa	
2 9	ओला	olaa	
3 0	खूबसूरत	xuubasuurata	

C-3 ヒンディー語（辞書表示あり）

出題したヒンディー語の検索ワードを AjaxIME を使って入力し、Google で検索し、その検索結果を記入してください。

注) スペースを入力したい場合だけは、アンダーバー “_” を入力してください。

【資料1】 Google 検索の方法、の「2. AjaxIME からの Google 検索」のように、検索を行い、結果を記録します。

ヒンディー語の転写規則は、【資料2】をご覧ください。

【練習問題】

検索ワード	AjaxIME 入力	検索結果(件数)
बिच्छू	bicc'uu	
धोना	d'onaa	
फटना	p'aTanaa	
चूतड़	cuutaRa	
गुड्डा	guDDaa	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	AjaxIME 入力	検索結果(件数)
1	कहानी	kahaanii	
2	पुदीना	pudiinaa	
3	खीरा	k'iiraa	
4	प्रणाली	praNaalii	
5	अंगूठी	AguuT'ii	
6	गाड़ी	gaaRii	
7	नौसेना	nausena	
8	कुर्सी	kursii	
9	निराला	niraalaa	
1 0	पिछला	pic'alaa	
1 1	निमंत्रण	nimaMtraNa	
1 2	नाम	naama	
1 3	पोस्टर	posTara	
1 4	बैंक का एकाउंटेंट	baiMka_kaa_ekaauMTeMTa	
1 5	चलना	calanaa	
1 6	बाज़ीगर	baaziigara	
1 7	पीलिया	piiliyaa	
1 8	पंडा	paMDaa	
1 9	चरखा	carak'aa	
2 0	किरण	kiraNa	
2 1	देर	dera	
2 2	उछालना	uc'aalanaa	
2 3	बाप	baapa	
2 4	चौड़ाई	cauRaaii	
2 5	पोता	potaa	
2 6	छल्ला	c'allaa	
2 7	पक्षपात	pak\$apaata	

2 8	क्रिया	kriyaa	
2 9	ओला	olaa	
3 0	खूबसूरत	xuubasuurata	

D-1 ポルトガル語

出題したポルトガル語の検索ワードをソフトキーボードを使って入力し、Google で検索し、その検索結果を記入してください。

注) 出題したポルトガル語については、つづりを正確に入力してください。

【資料1】Google 検索の方法、の「3. Soft Keyboard を用いた入力」のように、検索を行い、結果を記録します。

キー配列については、【資料3】ソフトキーボードのキー配列、をご確認ください。

【練習問題】

ポルトガル語特有の文字を入力してみましょう。

注) アルファベットの上の記号は、アルファベットを入力する前に入力します。

例) á の場合

「´」を入力してから「a」を入力します。

á ã ç Á é ô

検索ワード	検索結果(件数)
vale postal	
recibo	
nome verdadeiro	
comediante	
enxaguar	
já	
uma pitada	
roubo	
planície	
muito mais	

【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	検索結果(件数)
1	extraordinário	
2	greve	
3	gene	
4	o outro	
5	desgastar-se	
6	expectativa	
7	gás	
8	lenha	
9	não se dar bem com	
1 0	solitário	
1 1	malha	
1 2	lugar	
1 3	caderno de esboços	
1 4	chocar-se com	
1 5	bisneta	
1 6	guarda	
1 7	Sudeste da Ásia	
1 8	de uma vez	
1 9	destino	
2 0	magia	
2 1	ovário	
2 2	consignar	
2 3	o eu	
2 4	reservado	
2 5	esqui	
2 6	sol da meia-noite	
2 7	dominar	

2 8	à vontade	
2 9	média	
3 0	pôr carta no correio	

D-2 ポルトガル語

出題したポルトガル語の検索ワードを AjaxIME を使って入力し、Google で検索し、その検索結果を記入してください。

注) スペースを入力したい場合だけは、アンダーバー “_” を入力してください。

【資料1】 Google 検索の方法、の「2. AjaxIME からの Google 検索」のように、検索を行い、結果を記録します。

ポルトガル語の入力規則は、【資料2】をご覧ください。

【練習問題】

検索ワード	検索結果(件数)
vale postal	
recibo	
nome verdadeiro	
comediante	
enxaguar	
já	
uma pitada	
roubo	
planície	
muito mais	

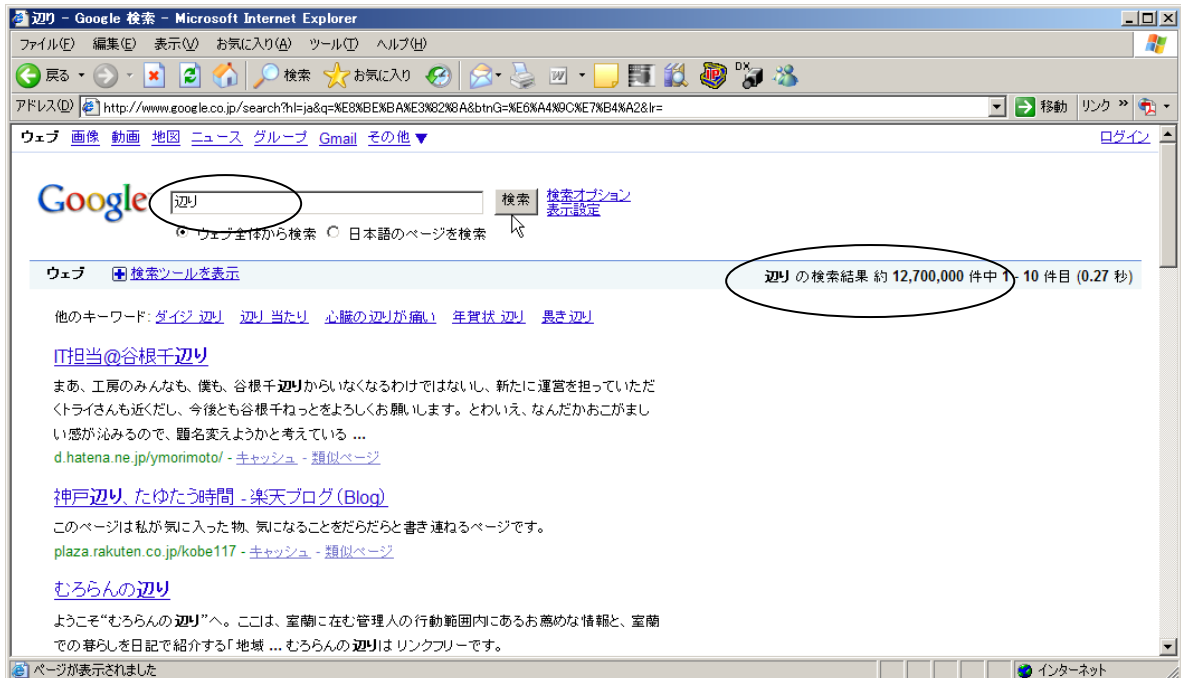
【本番問題】 スタートの指示が出てから検索を始めてください。

No.	検索ワード	検索結果(件数)
1	extraordinário	
2	greve	
3	gene	
4	o outro	
5	desgastar-se	
6	expectativa	
7	gás	
8	lenha	
9	não se dar bem com	
1 0	solitário	
1 1	malha	
1 2	lugar	
1 3	caderno de esboços	
1 4	chocar-se com	
1 5	bisneta	
1 6	guarda	
1 7	Sudeste da Ásia	
1 8	de uma vez	
1 9	destino	
2 0	magia	
2 1	ovário	
2 2	consignar	
2 3	o eu	
2 4	reservado	
2 5	esqui	
2 6	sol da meia-noite	
2 7	dominar	

2 8	à vontade	
2 9	média	
3 0	pôr carta no correio	

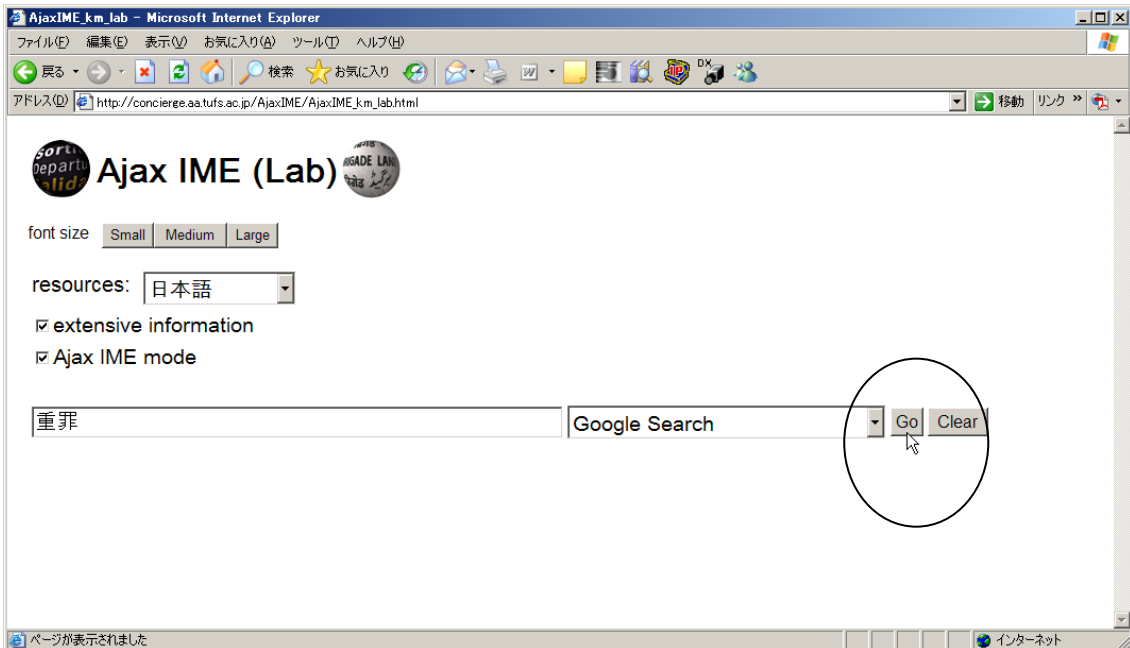
【資料1】 Google 検索の方法

1. Google のページに直接文字入力する場合

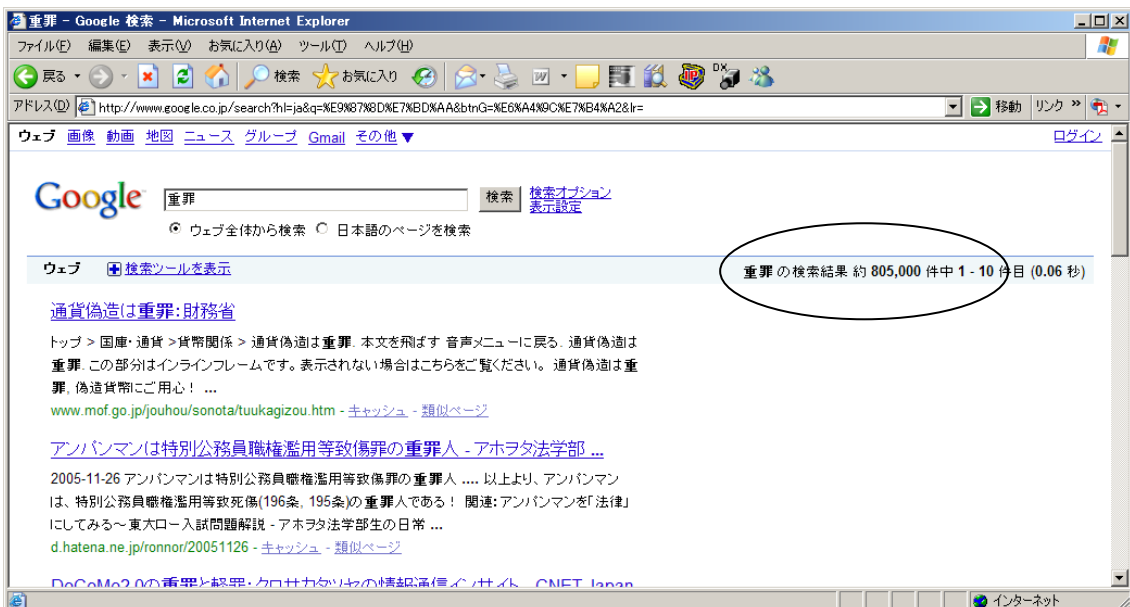


- (1) 上図のキーワード入力フレーム内にキーワードを入力ください。
- (2) 検索ボタンをクリックください。
- (3) 検索結果と検索ヒット件数が表示されます。ヒット件数を指定の用紙にメモしてください。
- (4) (1) から (3) を繰り返してください。

2. AjaxIME からの Google 検索

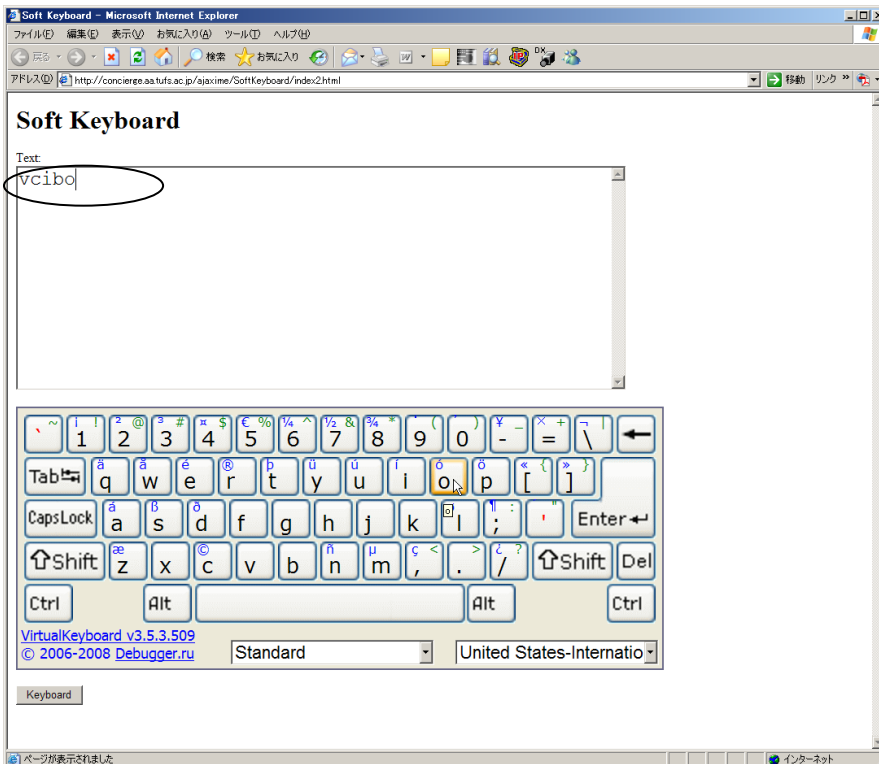


- (1) キーワード入力後に「Go」ボタンをクリックください。
- (2) 別ウィンドウが立ち上がって google 検索サイトでの検索結果が表示されます。
- (3) 検索結果のヒット件数が表示されます。ヒット件数を指定の用紙にメモしてください。

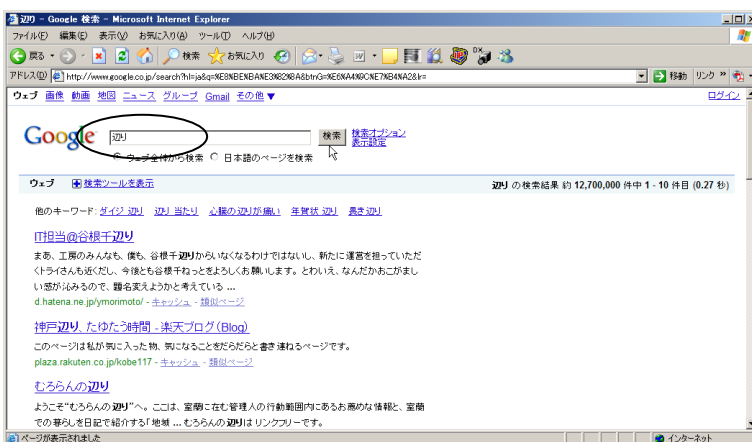


- (4) google 検索サイトのウィンドウを閉じてください。
- (5) (1) から (4) を繰り返してください。

3. Soft Keyboard を用いた入力



- (1) Soft Keyboard の画面上で、カーソルは Text フレームにおいてください。
- (2) 画面上のキーボードをマウスでクリックして入力ください。
- (3) 入力した文字が Text フレームエリアに表示されます。
- (4) 表示された文字をマウスでドラッグして、コピーしてください。
- (5) Google 検索画面の検索キーワード入力フレーム内にペーストしてください。
- (6) 検索をクリックし、検索結果が表示された画面のヒット件数を指定の用紙にメモしてください。



- (7) (1) から (6) を繰り返してください。

【資料2】 AjaxIME 入力規則

1. 共通ルール

転写規則に従ってローマ字入力を行い、入力したい語をエンターで確定します。
スペースを入力したい場合だけは、アンダーバー “_” を入力してください。

例えば、

long sleeves

を入力する場合は

long_sleeves

と入力してください。

2. 日本語の転写規則

(1) 転写規則はヘボン式です。

(2) 「ん」は、「n」を1回だけ入力ください。

ヘボン式

あ	A	い	I	う	U	え	E	お	O
か	KA	き	KI	く	KU	け	KE	こ	KO
さ	SA	し	SHI	す	SU	せ	SE	そ	SO
た	TA	ち	CHI	つ	TSU	て	TE	と	TO
な	NA	に	NI	ぬ	NU	ね	NE	の	NO
は	HA	ひ	HI	ふ	FU	へ	HE	ほ	HO
ま	MA	み	MI	む	MU	め	ME	も	MO
や	YA	い	I	う	U	え	E	お	O
ら	RA	り	RI	る	RU	れ	RE	ろ	RO
ん	N(M)								
が	GA	ぎ	GI	ぐ	GU	げ	GE	ご	GO
ざ	ZA	じ	JI	ず	ZU	ぜ	ZE	ぞ	ZO
だ	DA	ぢ	J I	づ	ZU	で	DE	ど	DO
ば	BA	び	BI	ぶ	BU	べ	BE	ぼ	BO
ぱ	PA	ぴ	P I	ぷ	PU	ぺ	PE	ぽ	PO
きゃ	KYA	きゅ	KYU	きょ	KYO				
しゃ	SHA	しゅ	SHU	しょ	SHO				
ちゃ	CHA	ちゅ	CHU	ちょ	CHO				
にゃ	NYA	にゅ	NYU	にょ	NYO				
ひゃ	HYA	ひゅ	HYU	ひょ	HYO				
みゃ	MYA	みゅ	MYU	みょ	MYO				
りゃ	RYA	りゅ	RYU	りょ	RYO				
ぎゃ	GYA	ぎゅ	GYU	ぎょ	GYO				
じゃ	JA	じゅ	JU	じょ	JO				
びゃ	BYA	びゅ	BYU	びょ	BYO				
ぴゃ	PYA	ぴゅ	PYU	ぴょ	PYO				

4. ポルトガル語の転写規則

(1) 記号は入力不要

é ú ó

など、上部に付く記号は AjaxIME では入力不要

e u o

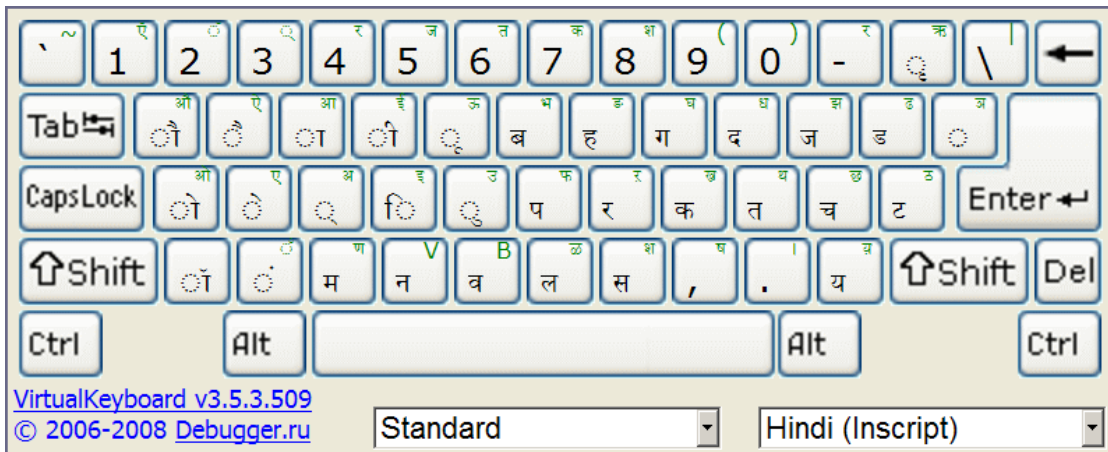
と入力してください。

【資料3】 ソフトキーボードのキー配列

1. ヒンディー入力用

Hindi(Inscript)

(1) Normal (初期状態)



黒字の文字をクリックして入力します。

緑色の文字を入力したいときは、Shift を押してください。

(2) Shift を押した場合



黒字の文字をクリックして入力します。

一字入力すると、(1) の状態に戻ります。

2. ポルトガル入力用

Portuguese

(1) Normal (初期状態)



黒字の文字をクリックして入力します。

緑色の文字を入力したいときは、Shift を押してください。

青色の文字を入力したいときは、Ctrl (または Alt) を押してください。

(2) Shift を押した場合。



(3) Ctrl (または Alt) を押した場合



(2) (3) では一字入力すると、(1) の状態に戻ります。

付録2 JIS X 0208:1997 から JIS X 0213:2004 で例示字形が変更された
字形リスト

No	x0208	x0213 current	unicode	current GID	IVS GID	IVS	備考
1	逢	逢	U+9022	7690	899	E0100	
2	芦	芦	U+82A6	7541	908	E0100	
3	飴	飴	U+98F4	7342	917	E0100	
4	溢	溢	U+6EA2	7343	968	E0100	
5	茨	茨	U+8328	7542	971	E0100	
6	鰯	鰯	U+9C2F	7636	973	E0100	
7	淫	淫	U+6DEB	7345	982	E0100	
8	迂	迂	U+8FC2	7346	994	E0100	
9	厩	厩	U+53A9	12156	1009	E0100	
10	樽	樽	U+5642	7347	1013	E0100	
11	餌	餌	U+990C	8515	1018	E0100	
12	襖	襖	U+8956	7349	1086	E0100	
13	迦	迦	U+8FE6	7351	1142	E0100	
14	牙	牙	U+7259	7544	1149	E0100	
15	廻	廻	U+5EFB	8516	1164	E0100	明朝のみ
16	恢	恢	U+6062	12154	1168	E0100	
17	晦	晦	U+6666	7352	1174	E0100	
18	葛	葛	U+845B	7354	1247	E0100	
19	鞆	鞆	U+9784	7355	1255	E0100	
20	釜	釜	U+91DC	12175	1260	E0100	明朝のみ
21	翰	翰	U+7FF0	7357	1311	E0100	
22	翫	翫	U+7FEB	7358	1335	E0100	
23	徽	徽	U+5FBD	7359	1371	E0100	
24	祇	祇	U+7947	7360	1392	E0100	

25	汲	汲	U+6C72	7545	1426	E0100	
26	灸	灸	U+7078	12155	1428	E0100	明朝・ゴシック 両方
27	笈	笈	U+7B08	12148	1432	E0100	
28	卿	卿	U+537F	7362	1464	E0101	
29	饗	饗	U+9957	7546	1488	E0100	
30	僅	僅	U+50C5	7363	1501	E0100	
31	喰	喰	U+55B0	7365	1538	E0100	
32	櫛	櫛	U+6ADB	7366	1545	E0101	
33	屑	屑	U+5C51	7367	1547	E0100	
34	糸	糸	U+7C82	12157	1557	E0100	明朝・ゴシック 両方
35	祁	祁	U+7941	7368	1571	E0100	
36	隙	隙	U+9699	12158	1616	E0100	
37	倦	倦	U+5026	7370	1629	E0100	
38	捲	捲	U+6372	7371	1642	E0100	
39	鍵	鍵	U+9375	12161	1658	E0100	明朝のみ
40	諺	諺	U+8AFA	7373	1675	E0100	
41	巷	巷	U+5DF7	7374	1747	E0100	
42	梗	梗	U+6897	12164	1764	E0100	明朝のみ
43	鵠	鵠	U+9D60	7377	1820	E0100	
44	甌	甌	U+7511	12169	1825	E0100	
45	叉	叉	U+53C9	12166	1851	E0100	明朝のみ
46	榭	榭	U+698A	7378	1901	E0100	
47	薩	薩	U+85A9	7379	1931	E0100	
48	鯖	鯖	U+9BD6	7380	1934	E0100	
49	鯖	鯖	U+9306	7381	1936	E0100	
50	鮫	鮫	U+9BAB	12165	1937	E0100	明朝のみ
51	餐	餐	U+9910	7547	1957	E0100	

52	杓	杓	U+6753	7383	2079	E0100	
53	灼	灼	U+707C	7384	2080	E0100	
54	酋	酋	U+914B	7386	2134	E0100	
55	楯	楯	U+696F	8510	2173	E0100	ゴシックのみ
56	薯	薯	U+85AF	7388	2194	E0100	
57	藪	藪	U+85F7	7389	2195	E0100	
58	哨	哨	U+54E8	7390	2211	E0100	
59	鞞	鞞	U+9798	7393	2274	E0100	
60	杖	杖	U+6756	12167	2289	E0100	明朝のみ
61	蝕	蝕	U+8755	7394	2310	E0100	
62	訊	訊	U+8A0A	8528	2354	E0100	
63	逗	逗	U+9017	7395	2364	E0100	
64	摺	摺	U+647A	7396	2396	E0100	
65	撰	撰	U+64B0	7398	2475	E0100	
66	煎	煎	U+714E	7399	2483	E0100	
67	煽	煽	U+717D	7548	2484	E0100	
68	穿	穿	U+7A7F	7549	2486	E0100	
69	箭	箭	U+7BAD	7550	2487	E0100	
70	詮	詮	U+8A6E	7400	2495	E0100	
71	噌	噌	U+564C	7401	2513	E0100	
72	遡	遡	U+9061	7402	2532	E0100	
73	揃	揃	U+63C3	7551	2605	E0100	
74	遜	遜	U+905C	7404	2611	E0100	
75	腿	腿	U+817F	7406	2642	E0100	
76	蛸	蛸	U+86F8	7408	2675	E0100	
77	辿	辿	U+8FBF	7409	2685	E0100	
78	樽	樽	U+6A3D	7410	2690	E0100	
79	歎	歎	U+6B4E	8536	2699	E0100	
80	註	註	U+8A3B	7412	2756	E0100	

81	瀦	瀦	U+7026	7413	2761	E0100	
82	抄	抄	U+6357	7414	2799	E0100	
83	槌	槌	U+69CC	7415	2810	E0100	
84	鎚	鎚	U+939A	7416	2812	E0100	
85	辻	辻	U+8FBB	7691	2822	E0100	
86	挺	挺	U+633A	12171	2852	E0100	明朝のみ
87	鄭	鄭	U+912D	7419	2866	E0100	
88	擢	擢	U+64E2	7420	2871	E0100	
89	溺	溺	U+6EBA	7421	2878	E0100	
90	兎	兎	U+514E	8537	2902	E0100	
91	堵	堵	U+5835	7424	2904	E0100	
92	屠	屠	U+5C60	7425	2907	E0100	
93	賭	賭	U+8CED	7426	2914	E0100	
94	滯	滯	U+701E	7429	3010	E0100	
95	遁	遁	U+9041	12172	3017	E0100	
96	謎	謎	U+8B0E	7430	3028	E0100	
97	灘	灘	U+7058	7431	3029	E0100	
98	檣	檣	U+6962	7432	3032	E0100	
99	禰	禰	U+79B0	7433	3061	E0100	
100	牌	牌	U+724C	7435	3107	E0100	
101	這	這	U+9019	7436	3123	E0100	
102	秤	秤	U+79E4	7437	3125	E0100	
103	駁	駁	U+99C1	12173	3145	E0100	明朝のみ
104	箸	箸	U+7BB8	7439	3150	E0100	
105	叛	叛	U+53DB	7552	3178	E0100	
106	挽	挽	U+633D	7442	3198	E0100	
107	誹	誹	U+8AB9	8511	3226	E0100	
108	樋	樋	U+6A0B	7443	3231	E0100	
109	稗	稗	U+7A17	7444	3243	E0100	

110	逼	逼	U+903C	7445	3255	E0100	
111	謬	謬	U+8B2C	7446	3261	E0100	
112	豹	豹	U+8C79	12147	3271	E0100	
113	廟	廟	U+5EDF	7447	3272	E0100	
114	瀨	瀨	U+7015	7448	3286	E0100	
115	斧	斧	U+65A7	12174	3304	E0100	明朝のみ
116	蔽	蔽	U+853D	7450	3369	E0100	
117	瞥	瞥	U+77A5	7451	3379	E0100	
118	蔑	蔑	U+8511	8543	3380	E0100	
119	篇	篇	U+7BC7	7553	3385	E0100	
120	媿	媿	U+5A29	7452	3392	E0100	
121	鞭	鞭	U+97AD	12176	3394	E0100	明朝のみ
122	庖	庖	U+5E96	7453	3423	E0100	
123	蓬	蓬	U+84EC	7454	3437	E0100	
124	鱒	鱒	U+9C52	7456	3508	E0100	
125	迄	迄	U+8FC4	7554	3516	E0100	
126	儲	儲	U+5132	7458	3579	E0100	
127	餅	餅	U+9905	7459	3585	E0100	
128	粩	粩	U+7C7E	12178	3588	E0100	
129	爺	爺	U+723A	12179	3598	E0100	明朝のみ
130	鎧	鎧	U+9453	7460	3612	E0100	
131	愈	愈	U+6108	7461	3614	E0100	
132	猷	猷	U+7337	8547	3634	E0100	
133	漣	漣	U+6F23	7463	3800	E0101	
134	煉	煉	U+7149	7464	3801	E0101	
135	簾	簾	U+7C3E	12150	3802	E0100	
136	榔	榔	U+6994	7465	3821	E0100	
137	冤	冤	U+51A4	7467	3994	E0100	
138	叟	叟	U+53DF	8552	4098	E0100	

139	咬	咬	U+54AC	12162	4134	E0100	明朝のみ
140	嘲	嘲	U+5632	7468	4199	E0100	
141	囀	囀	U+56C0	8554	4220	E0100	
142	徘	徘	U+5F98	8512	4562	E0100	
143	扁	扁	U+6241	7555	4709	E0100	
144	狡	狡	U+72E1	12163	5395	E0100	明朝のみ
145	甕	甕	U+7515	12152	5470	E0100	
146	甦	甦	U+7526	12168	5473	E0100	明朝のみ
147	疼	疼	U+75BC	12146	5507	E0100	
148	竈	竈	U+7AC8	12170	5703	E0100	
149	筵	筵	U+7B75	12151	5742	E0100	明朝のみ
150	箒	箒	U+7BDD	7556	5767	E0100	
151	隼	隼	U+8171	12160	6024	E0100	明朝のみ
152	艘	艘	U+8258	8568	6076	E0100	
153	芒	芒	U+8292	12177	6091	E0100	
154	虔	虔	U+8654	12159	6247	E0100	明朝のみ
155	蜃	蜃	U+8703	12181	6276	E0100	明朝・ゴシック 両方
156	蠅	蠅	U+8805	12180	6303	E0100	
157	訝	訝	U+8A1D	12153	6428	E0100	
158	鞞	鞞	U+9771	8513	6918	E0100	明朝のみ
159	騙	騙	U+9A19	7557	7014	E0100	
160	鴉	鴉	U+9D09	8575	7128	E0100	
161	摯	摯	U+646F	12149	4786	E0100	明朝・ゴシック 両方
162	寛	寛	U+5BDB	1284	7813	E0101	
163	朗	朗	U+6717	3819	7864	E0102	
164	菴	菴	U+8612	9202	7983	E0101	
165	諶	諶	U+8B7F	12204	7996	E0100	

166	懲	懲	U+61F2	2778	8480	E0101	
167	悞	悞	U+609E	12206	8766	E0100	明朝・ゴシック 両方
168	焯	焯	U+7152	12208	8945	E0100	明朝・ゴシック 両方
169	驩	驩	U+9A4A	12209	9391	E0100	明朝・ゴシック 両方
170	夆	夆	U+5906	12205	10483	E0100	明朝・ゴシック 両方
171	嘆	嘆	U+66B5	12207	10769	E0100	