

内容梗概

本研究は Z39.50 検索クライアントを開発するとともに、検索要求にあったデータベースを選択するための支援機能を実現した。

Z39.50 は検索の際のクライアントとサーバ間で通信されるデータ構造やデータ処理方式について取り決めた ISO 及び JIS の規格である。Z39.50 に基づいてシステムを構築することにより、一つのクライアントから複数のサーバに接続できるため、利用者は異なるデータベースを透過的に検索できる。一方、World Wide Web (以下 WWW と呼ぶ) がインターネット上のハイパーテキストの概念に基づく分散型情報システムとして登場し、爆発的に普及した。現在、アメリカ議会図書館、OCLC をはじめ、アメリカやヨーロッパの図書館システムを中心に数多くの Z39.50 サーバが構築され、データベースは増加の一途をたどっている。そのため、Z39.50 によってデータベース選択の幅は広がったものの、検索要求にあったデータベースを見つけることは困難になった。

利用者が適切なデータベースを選択できるためには、データベース接続情報を網羅的に収集し、組織化してクライアントに登録する必要がある。しかしながら、データベース接続情報を収集し公開している WWW サイトや既存クライアントは、どれもデータベース接続情報を独自の形式で記述しているため、それぞれの形式を理解しなければならず、その大部分はデータベース名ごとに並び替えてあるだけなので、利用者自身で組織化しなければならない。また、全てのデータベース接続情報をひとつのクライアントが収集し組織化することは難しい。

この問題は利用者がそれぞれ収集し組織化したデータベース接続情報をお互いに補完し共有できるようになることで解決すると考えた。そこで、本研究では個人や組織が作成したコンテンツをお互いに利用できる WWW に着目し、データベースに接続するための情報を WWW で共有し、Z39.50 でデータベースにアクセスし、検索するシステムを開発した。

本システムの構成要素は以下の 3 点である。

1. Z39.50 用 URL によるデータベース接続情報
2. Z39.50 用 URL を扱える WWW ブラウザ
3. その WWW ブラウザから呼び出せる Z39.50 クライアント

Z39.50 用 URL は、RFC で提案された Z39.50 サーバを一意に表すための URL である。Z39.50 用 URL でデータベース接続情報を記述することにより、全てのデータベース接続情報を同一形式で記述でき、かつデータベースを一意に識別できるため、WWW ブラウザからデータベースに直接アクセスすることが容易になる。しかし、Z39.50 用 URL を解釈できる WWW ブラウザは確認出来なかったため、本研究では新たに WWW ブラウザを開発した。この WWW ブラウザと Z39.50 クライアントにより、利用者は通常の WWW ページを閲覧したり、リンクをたどるのと同じ操作で、Z39.50 データベースを検索できる。具体的には、リンク集やディレクトリ型検索エンジンを利用して WWW ページを閲覧するのと同様に、すでに収集し組織化されたデータベース接続情報を利用して検索できる。

また、本研究ではデータベースに関する情報を収集し WWW 上で公開しているサイトから HTML 文書を取り寄せ、その記述を解析してデータベース接続情報を抜き出し Z39.50 用 URL に変換した。さらに、データベース接続情報の組織化も行った。組織化としては頻繁に利用するデータベースリストの作成や、分野ごとにデータベース接続情報を自動分類した。分野ごとの自動分類はデータベースのインデックス語とヒット件数を取り寄せる Z39.50 の Scan 機能を利用して行った。自動分類の利点は人手で分類するよりも、素早くデータベースを分類し追加できる点にある。

本研究では、Z39.50 と WWW の特性を活かした新しいシステム形態の可能性を示したとともに、個別に収集し組織化されていたデータベース接続情報を収集・加工・流通・利用というサイクルの中で共有できるようになった。

Synopsis

I implemented a Z39.50 client containing a facility to support database choice for users' demand for a search.

Z39.50 is an ISO and JIS standard, and specifies a client/server based protocol for information retrieval. It specifies procedures and structures for a client to search a database provided by a server, to retrieve database records identified by a search, and to scan a term list. It is possible to search transparently more than one Z39.50 server from one client. On the other hand, World Wide Web, which is based on the idea of the hypertext distributed information system on Internet, is spreading explosively. A lot of Z39.50 servers has been built including the Library of Congress, OCLC around the library system in United States and Europe, and databases fall into the increase. Because of that, though the width of the database choice spread out by Z39.50, it became difficult to find a database in a demand for a search.

Information for database connection is collected exhaustively and is organized, and users must register the information for the client in order to choose the suitable database. There are WWW sites which collect the information for database connections and introduced to the public. However, because all the information for database connections described with the individual form in these site and clients, users must understand each form. Moreover, users must organize the information for database connections by a user himself because the most part of the WWW site and the client is just arranged in database name of the order. Moreover, it is difficult for only one client to collect all information for database connection and to organize it.

These problems can be solved by complementing the information for database connection that a user collected them respectively and which was organized to each other and sharing it. In this research, it paid attention to WWW that each other could share the contents which an individual and organization made. This system which the information was shared with WWW, and database were accessed and searched with Z39.50 was developed.

The components of this system are three points of the following.

1. Information for database connection by the URL for Z39.50
2. The WWW browser that can support the URL for Z39.50
3. The Z39.50 client, which can be launched from that browser

The URL for Z39.50, proposed with RFC, is a mechanism to identify a unique Z39.50 server. Using the URL for Z39.50, all information for database connection can be described in the same form and can be uniquely identified a database. So it becomes easy to access a database directly from the WWW browser. But, it was developed by this research newly because a WWW browser that could support URL for Z39.50 didn't exist. With this WWW browser and the Z39.50 client, users can search Z39.50 databases by the operation which is the same as browsing usual WWW pages and following links. For example, as users browse WWW pages with a link page and a directory-type search engine, users can search databases with information for database connection to collect already and which is organized.

Moreover, it obtained a HTML document by this research from the WWW site where information about the database was collected. Then, that description was analyzed, and information for database connection was picked out, and it converted into the URL for Z39.50. Furthermore, the organization of the information for database connection was done, too. The gathering of a database list to use frequently and the automatic classification

of the information for database connection information of every field were done. The automatic classification of every field used the Scan facility on Z39.50 that can fetch ordered index words of the database and the number of hit. The advantage of the automatic classification is quick creation of the list rather than it is classified by hands.

I showed that there is a strong potential for a profitable and synergistics relationship between the WWW and Z39.50. And by this research, it became to share information, which respectively collected and organized, in the cycle of the collection, processing, circulation, and use.

Z39.50 におけるデータベース選択支援機能の実現

江草由佳

2000年3月

目次

| | | |
|-------|-----------------------|----|
| 第1章 | はじめに | 1 |
| 1.1 | Z39.50 と WWW | 1 |
| 1.2 | Z39.50 におけるデータベース選択 | 2 |
| 第2章 | データベース選択支援システム | 4 |
| 2.1 | システムの概要 | 4 |
| 2.2 | システムの構成 | 5 |
| 2.3 | データベース接続情報の収集 | 6 |
| 2.4 | データベース接続情報の加工 | 6 |
| 2.4.1 | 自動分類の定式化 | 7 |
| 2.5 | データベース接続情報の流通 | 7 |
| 2.6 | データベース接続情報の利用 | 8 |
| 2.6.1 | WWW ブラウザ | 9 |
| | システム構成 | 9 |
| | 機能 | 9 |
| 2.6.2 | Z39.50 クライアント | 9 |
| | 構成 | 9 |
| | 機能 | 10 |
| 2.7 | システムの実行例 | 11 |
| 第3章 | 考察 | 18 |
| 3.1 | 既存のデータベース選択支援システムとの比較 | 18 |
| 3.2 | Z39.50 と WWW との関わり | 19 |
| 第4章 | おわりに | 21 |
| | 謝辞 | 22 |
| | 参考文献 | 23 |
| | 参考文献 | 25 |
| | 付録 | 26 |

| | | |
|------|--|-----|
| 付録 A | データベースリスト | 27 |
| A.1 | お気に入りのデータベースリスト | 27 |
| A.2 | 分野事に自動分類したデータベースリスト | 28 |
| A.3 | データベース名ごとに並び替えたデータベースリスト | 30 |
| 付録 B | データベース接続情報を抽出するプログラム | 50 |
| B.1 | C シェルプログラム コード名: addNewServer | 50 |
| B.2 | CSV ファイルに変換するプログラム コード名: lineanalysis.pl | 52 |
| 付録 C | WWW ブラウザのプログラム | 53 |
| 付録 D | Z39.50 クライアントのプログラム | 58 |
| D.1 | client.c | 59 |
| D.2 | client.tcl | 91 |
| D.3 | Makefile | 103 |
| D.4 | kappa | 113 |

目次

| | |
|---|----|
| 2.1 システムの全体図 | 5 |
| 2.2 接続ウィンドウの機能 | 10 |
| 2.3 検索ウィンドウの機能 | 10 |
| 2.4 WWW ブラウザ初期画面 | 11 |
| 2.5 分野毎に自動分類したデータベースリスト | 12 |
| 2.6 サーバごとに並び変えたデータベースリスト | 12 |
| 2.7 良く使うデータベースリスト | 13 |
| 2.8 Z39.50 クライアントの起動画面 | 13 |
| 2.9 Z39.50 クライアントの検索画面 | 14 |
| 2.10 複数データベース検索実行例 | 15 |
| 2.11 ドキュメント表示例 | 15 |
| 2.12 検索ウィンドウ：Japan/MARC サーバ | 16 |
| 2.13 検索ウィンドウ：図書館情報大学デジタル図書館システム Z39.50 サーバ | 16 |
| 2.14 検索ウィンドウ：University of Wisconsin, Madison | 17 |
| 3.1 WWW-Z39.50 ゲートウェイシステムとプロトコルとの関係 | 19 |
| 3.2 本システムのプロトコルとの関係 | 20 |

第 1 章

はじめに

1.1 Z39.50 と WWW

1960 年代後半からオンライン検索システムの開発が始まり、1972 年には DIALOG や ORBIT などがデータベースベンダとして多くの商用データベースを抱え、オンラインサービスを開始した。時同じくしてアメリカ議会図書館、OCLC、RLIN などが機械可読目録 (MARC) のデータベースを構築し、書誌ユーティリティとしての機能を果たし始めた。これらのシステムは公衆回線を通じて端末からホストコンピュータにアクセスする中央集中型システムであり、この時代はこの形態のシステムが情報を得る上で最も効率がよかった。

しかしながら、中央集中型システムといえども全ての情報が集中しているわけではないため、様々な情報要求を満たすためには複数のシステムを利用せざるを得なかった。ところが、ここで問題となったのはコストと使い勝手であった。すなわち、これらのデータベースを使用するにはそれぞれ高額の使用料を支払わなければならない、そのうえ、複数のデータベースの使用料に見合う情報は必ずしも得られなかった。特に書誌ユーティリティにおいては重複データが数多くあり、情報を得る上で効率が悪く、使い勝手の面でもシステムごとに接続手順や検索コマンドに違いがあり、利用者の検索要求を十分満たせないという状況であった。

こういう状況の下、1979 年に主要な書誌ユーティリティ間で目録データを交換し、互いのデータを透過的に検索できることを目指したプロジェクトが開始され、その後、OSI の規格と平行して ANSI の構成団体である NISO で議論され、1988 年に標準情報検索プロトコル Z39.50 として規格が制定された [1][2]。

Z39.50 の動きとは別にインターネットが 1982 年頃から普及し始め [3]、1989 年には World Wide Web (以下 WWW と呼ぶ) というハイパーテキストの概念に基づく分散型情報システムが登場した [4]。WWW は、CERN (Organization[旧 Conseil]Européenne pour la Recherche Nucléaire) の Tim Berners-Lee 氏らによって、研究者の間での効率的な情報共有を目指して開発されたものである [5]。1993 年に米国の NCSA (National Center for Supercomputing Applications) でブラウザ Mosaic が開発されると WWW は一気に普及していった。それは Mosaic が、わかり易い Graphical なユーザ・インターフェースのもとに、クライアントの機能を統合し、他のブラウザとは一線を画する機能と洗練度があったため、ユーザは Mosaic の使い方だけを覚えれば、さまざまなリソースにアクセスできるようになったからである [5]。

このように急速に WWW が広がる中で Z39.50 は独自に普及を続け、1990 年代以降、アメリカ議会図書館、OCLC をはじめ、アメリカ、ヨーロッパで図書館システムを中心に数多くの Z39.50 検索システムが実装されるようになり [6][7][8]、2000 年 1 月現在、Index Data 社が作成しているサーバリスト [9] に登録されているデータベースだけでも 381 にのぼる。

日本でも欧米の動きに呼応するように、1999年にはJIS規格として採用され[10]、デジタル図書館のプロジェクト[11][12][13]を中心にZ39.50で検索を提供するシステムが構築されるようになってきた。一方、WWWも指数関数的な増大を続けており、インターネットに接続さえすればさまざまな情報にアクセスできるという状況になった[14]。

このように現在、Z39.50とWWWが両方インターネット上で利用できるようになったが、この2つのシステムは性格が全く異なる。すなわち、Z39.50[15]は情報検索のために標準化されたプロトコルで、クライアント・サーバ環境において、クライアントとサーバが通信する際のデータ構造やデータ変換の処理方式を規定したプロトコルであり、システム内部でのデータ構造や検索処理がシステムごとに異なっても、通信する際に同じ構造と処理方式なのでそれぞれのシステム構成を維持したまま互いのシステムを利用でき、Z39.50を用いたシステムは情報検索システムである。また、WWWはインターネット上でファイルの送受信を行うためのプロトコルHTTP(HyperText Transfer Protocol)と、機種に依存しないハイパーテキストのための文書形式HTML(HyperText Markup Language)と、インターネット上に存在する様々なリソースをあらわすためのアドレスURL(Uniform Resource Locator)[14]を利用してさまざまな異なるシステム間でデータのやり取りを可能にしている情報共有のための情報システムである。WWWは情報を共有すること、すなわち、個人や組織が作成したコンテンツをお互いに利用できる点に優れ、Z39.50ではさまざまな機関や組織が提供しているデータベースを共通の手順で検索できるなど、透過的な情報検索を行える点で優れている。

このようなWWWとZ39.50の特性を活かして2つのシステムをさらに活用できるという提案もある[16]が、実装されているシステムは確認されていない。そこで、本研究では、情報共有に適しているWWWの仕組みを利用して、データベースに接続するための情報を共有し、情報検索に適しているZ39.50を使ってデータベースにアクセスし検索するシステムを開発した。

1.2 Z39.50におけるデータベース選択

Z39.50において、検索要求にあったデータベースを選択し検索するためには、以下の2点が必要である。

1. データベース接続情報をクライアントへ登録

Z39.50ではデータベースに接続するための手法は標準化されているのでZ39.50クライアントがあれば世界中のZ39.50データベースを検索できる。しかし、データベースに接続するための情報(以下データベース接続情報と呼ぶ)を知らなければデータベースに接続できない。そこで、利用者はデータベース接続情報をどこかで手に入れて自分でクライアントに入力しなければならない。

2. データベースの組織化

データベースの内容が分からなければ、検索要求にあったデータベースが選択できないので、データベースの扱っている分野など、データベースの内容がわかりやすいように組織化されていなければならない。

しかし、以下のような問題がある。

1. データベース接続情報の入手が困難

BookWhere2000[17]やWillow[18]のようにデータベース接続情報をあらかじめ登録しているクライアントもあるが、データベース全てが登録されているわけではないので、登録されていないデータベースは、データベース接続情報を入手しなければならない。入手する方法としては、WWW収集サイト[9][19][20][21]と他のクライアント、データベースを提供しているサイトのから取得があるが、どれもデータベース接続情報を独自の形式で記述しているので、それぞれの形式を理解しなければ、データベース接続情報を得ることはできない。

2. データベース接続情報の組織化を自分でやらなくてはならない。

Cornell 大学の Gateway システム [22][23] のように WWW ゲートウェイを利用したデータベースの提供を行っているところでは、分野ごとにデータベースをまとめるなどのデータベースの組織化を行っているところもあるが、WWW ゲートウェイを公開する機関が提供するデータベースにしか検索できない。そのため、WWW ゲートウェイで提供していないデータベースを検索するには、クライアントで検索する必要があるが、クライアントや WWW 上のデータベース接続情報収集サイトや既存のクライアントの大部分は、データベース名ごとに並び替えてあるだけなので、各ユーザが自分でデータベースを組織化しなければならない状況である。

これらの問題は、クライアントや WWW ゲートウェイが、全てのデータベース接続情報を収集し組織化を行えば解決できるが、データベースが増加している現在、全てのデータベース接続情報を収集し組織化するのは不可能である。そこで、データベース接続情報をお互いに利用できるような枠組みが必要になってくる。収集したデータベース接続情報を相互に利用できれば、特定の収集サイトやクライアントが全てを収集しなくても済むし、組織化したデータベース接続情報を相互に利用できれば、得意な分野のデータベースだけの組織化を担当するなど、お互いに補完しあうことができるため、完全なデータベースの組織化を行う必要がなくなるからである。

そこで本研究では、作成したドキュメントをお互いに参照できる WWW の枠組みを利用したデータベース選択支援システムを開発した。このシステムは以下のことが可能である。

- 利用者がいちいち各 WWW サイトや各クライアントの記述を理解して、データベース接続情報の部分を抜き出さなくて良い。
- 各自が組織化したデータベース接続情報を相互に参照できる。
- 自分でクライアントにデータベース接続情報を登録しなくてもデータベースに接続し、検索できる。

第 2 章

データベース選択支援システム

2.1 システムの概要

本研究では、データベースの選択支援機能を持つシステムを開発した。データベース選択支援機能は、データベース接続情報の収集・加工・流通・利用を一連の流れで行えるシステムにより実現される。データベース接続情報の収集・加工・流通・利用とはそれぞれ以下の通りである。

データベース接続情報の収集 データベース接続情報を集めること。

具体的には、WWW 上のデータベース接続情報収集サイトから収集したり、データベースを提供している機関が作成しているドキュメントを調査して収集したり、クライアントがあらかじめ登録しているデータベース接続情報を収集したりすること。

データベース接続情報の加工 データベースの内容が分かるように組織化すること。

具体的には、良く使うデータベースをまとめたり、データベースが扱っている分野事に分類したり、データベース名や提供している期間毎に並び替えたりすること。

データベース接続情報の流通 収集し組織化されたデータベース接続情報を収集した人以外でも利用できるようになっていること。

具体的には WWW のように自分以外が作成したデータベースリストを自分が作成したのと同様に利用できる状態にデータベース接続情報になっていること。

データベース接続情報の利用 データベース接続情報を閲覧したり、データベース接続情報を使ってデータベースを検索すること。

具体的にはブラウザを使ってデータベース接続情報を閲覧したり、クライアントを使ってデータベースに接続して検索すること。

後の節でくわしく述べるが、ここでは、どのようにデータベース接続情報の収集・加工・流通・利用を一連の流れで行えるようにしたかについて簡単に説明する。以下の 3 点にまとめられる。

1. データベース接続情報を Z39.50 用 URL で表現した。
2. Z39.50 用 URL を扱える WWW ブラウザを開発した。
3. その WWW ブラウザから呼び出すことのできる Z39.50 クライアントを開発した。

URL とはインターネット上に存在する様々なリソースをあらわすためのアドレスであり、WWW で利用できる。そのため、データベース接続情報を Z39.50 用 URL で表現した HTML ファイルを WWW サーバ上に置けば、WWW

上であればどこでも参照できる。つまり、データベース接続情報の流通が行える。

1章で述べたように、データベースが増加している現在、特定の収集サイトやクライアントが全てのデータベース接続情報を収集することは不可能である。しかし、収集したデータベース接続情報を相互に利用できれば、全てを収集しなくても済む。また、URL という標準化された記述形式でデータベース接続情報が記述されているので、他のサイトが収集したデータベース接続情報から、データベース接続情報を取り出し、自分のデータベースリストに加えやすくなる。つまり、収集しやすくなる。よって、データベース接続情報の収集が行える。

また、組織化でも同様なことがいえる。様々な検索要求、データベースの増加により、完全なデータベースの組織化を行うことはできないが、組織化したデータベース接続情報を相互に利用できれば、得意な分野のデータベースだけの組織化を担当するなど、お互いに補完しあうことができる。つまり、データベース接続情報の加工が行える。

そして、Z39.50URL が扱える WWW ブラウザと WWW ブラウザから呼び出される Z39.50 クライアントがあるので、ユーザは、WWW ブラウザを使って、WWW サーフィンし、Z39.50 用 URL のリンクをたどるだけで、データベースに接続する Z39.50 クライアントで検索できる。つまり、データベース接続情報の利用が行える。

2.2 システムの構成

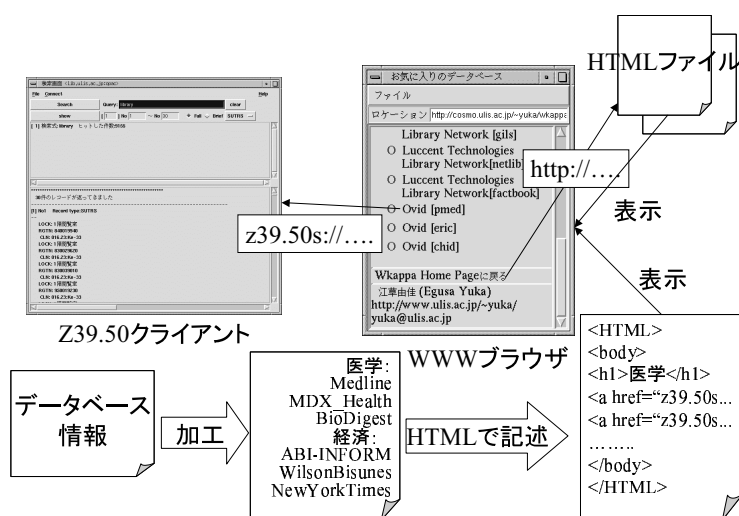


図 2.1: システムの全体図

本システムはデータベース接続情報を収集・加工・流通の部分であるデータ作成部、データベース接続情報の利用の部分である WWW ブラウザと Z39.50 クライアントで構成される。(図 2.1)

データ作成部では、データベース接続情報の収集・加工・流通・利用のサイクル中の収集・加工・流通の部分である。WWW ブラウザ、Z39.50 クライアントはデータベース接続情報の収集・加工・流通・利用のサイクル中の利用の部分である。

データベース接続情報の収集、データベース接続情報の加工、ヘルプなどのドキュメントの作成、HTML ファイルへの変換を行う。作成した HTML ファイルは WWW サーバ上に置き、インターネット上であれば、どこでも URL で閲覧できるようにした。以降より、データベース接続情報の収集・加工・流通・利用のサイクルの順を追ってシステ

ムの全体を説明する。

2.3 データベース接続情報の収集

データベース接続情報の収集とは、データベースに接続するためホスト名、データベース名、ポート番号を集めることであり、データベース情報の収集・加工・流通・利用のサイクル中の最初のステップである。

データベース接続情報はデータベースに関する情報を収集しているサイト [9] から取り寄せ収集する。このサイトではデータベース接続情報だけではなく、そのほかのデータベースに関する情報についても収集している。例えば、Title,author などアクセスポイントの種類、US-MARC や UNI-MARC やブレインテキストなど扱っているレコードフォーマットなどである。このサイトの対象は世界中で無料公開されているデータベースであり、収集データは3日に一度程度の頻度で更新されている。収集されたデータベースに関する情報は、データベースごとに HTML の表の一行で記述してある。

本研究では、このサイトがデータベースの収集数がほかの収集サイトに比べて多いこと、世界中のデータベースを対象にしていること、頻繁に更新していること、データベースにつき一行で HTML の table タグで記述されているためデータベース接続情報が抽出しやすいことなどの理由から、このサイトを本研究で利用するデータベース接続情報の収集サイトとした。

本研究では、このサイトから HTML ファイルを取得し、データベースごとに、データベース接続情報（ホスト名、ポート番号、データベース名）を抽出するプログラム（付録 B）を作成し、データベース接続情報の収集とした。

2.4 データベース接続情報の加工

データベース接続情報の加工とは、収集したデータベース接続情報を検索利用者が選択できるように組織化することであり、データベース情報の収集・加工・流通・利用のサイクル中の収集の次に行う 2 番目のステップである。

本システムでは、データベース接続情報の加工は以下の 3 つを行なった。

1. サーバごとに並び替えた全データベースリストの作成
2. 分野毎に自動分類したデータベースリストの作成
3. お気に入りデータベースリストの作成

この内、データベースの自動分類を行った理由としては、自動分類を行うことでユーザは、人手で分類するよりも早く新しく追加された分野で分類ずみのデータベースを知ることができ、また、人手で分類するよりもコストを押さえることができるからである。

また、自動分類は以下の方法で行なった。この方法をとった理由は、Z39.50 の機能に scan 機能とあって、データベースのインデックス語のブラウジング機能があり、この機能を利用すると、データベースが持っている全てのインデックスワードとそれに対応するヒット件数を取得することが可能であること、データベースのインデックス語とそれに対応するヒット件数はそのデータベースの分野を表していると筆者は考えたからである。

そこで、ある分野を代表するデータベースをあらかじめ選び、そのデータベースのインデックス語とそれに対応するヒット件数を基準にして、データベースを分野事に分類した。

1. データベース情報を公開しているサイト [9] からデータベース情報を取り寄せる。
2. データベース情報から scan 機能を持っているデータベースを選ぶ。

3. scan 機能を持っているデータベースすべてに対して、scan 機能を使ってすべてのインデックス語とインデックス語に対応するヒット件数を取得する。
4. すべてのデータベース同士の相関係数を求める。
5. ある分野を代表するデータベースを選ぶ。
6. ある分野を代表するデータベースとその他のデータベースとの相関係数をその分野と各データベースとの適合度とする。
7. ある分野との適合度が 0.6 以上のデータベースはその分野と判定する。

2.4.1 自動分類の定式化

今回はデータベースの全てのインデックス語とインデックス語に対応するヒット件数のピアソンの積率相関係数（以下：相関係数）を求めた。相関係数とは、2 変量間の直線的な関連の強さを表す尺度であり、-1 から 1 の間の値をとる。直線的な関連性が弱いときには、相関係数は 0 に近い値をとり、直線的な関連性が強いときには、-1 あるいは 1 に近い値をとる [24]。そこで、相関係数を求めることでデータベースとデータベースがどれくらい直線的な関連があるかどうかを知ることができる。

- Scan 機能を使ってインデックス語とヒット件数が取得できたデータベース $1, 2, \dots, i, \dots, j, \dots, D$ の D 個のデータベースを分野ごとに自動分類する。本報告では $D = 72$ のデータベースについて、分野ごとに自動分類を行った。
- データベース i と j に含まれる K_{ij} 個のインデックス語の集合を $T_{ij} = \{T_{ij1}, T_{ij2}, \dots, T_{ijl}, \dots, T_{ijK_{ij}}\}$ とする。
- インデックス語の集合 T_{ij} に対応するデータベース i のヒット件数の集合を $H_i = \{H_{i1}, H_{i2}, \dots, H_{il}, \dots, H_{iK_{ij}}\}$ とし、インデックス集合 T_{ij} に対応するデータベース j のヒット件数の集合を $H_j = \{H_{j1}, H_{j2}, \dots, H_{jl}, \dots, H_{jK_{ij}}\}$ とする。
- H_i の平均を $\widetilde{H}_i = \frac{\sum_{l=1}^{K_{ij}} H_{il}}{K_{ij}}$ 、 H_j の平均 $\widetilde{H}_j = \frac{\sum_{l=1}^{K_{ij}} H_{jl}}{K_{ij}}$ とする。

あるデータベース i とあるデータベース j の相関係数 r_{ij} は、

$$r_{ij} = \frac{\sum_{l=1}^{K_{ij}} (H_{il} - \widetilde{H}_i)(H_{jl} - \widetilde{H}_j)}{\sqrt{\sum_{l=1}^{K_{ij}} (H_{il} - \widetilde{H}_i)^2} \sqrt{\sum_{l=1}^{K_{ij}} (H_{jl} - \widetilde{H}_j)^2}}$$

である。

- 判定する分野の集合を $C = \{C_1, C_2, \dots, C_f, \dots, C_F\}$ （今回は $F=3$ ）とする。

ある分野 C_f を代表するデータベースを i としたとき、 i と j の相関係数 r_{ij} を j データベースの分野 C_f の適合度とし、適合度が $r_{ij} > 0.6$ であればデータベース j は分野 C_f とする。

2.5 データベース接続情報の流通

この節では、データベース情報の収集・加工・流通・利用のサイクル中の 3 番目のステップ、データベース接続情報の流通について述べる。データベース接続情報を流通させるには、流通できる仕組みが必要である。

そこで、データベース接続情報を誰もが利用できるためにデータベース接続情報を相互に利用できる WWW の仕組みを本研究では採用した。実際に行ったことは、データベース接続情報を Z39.50 用 URL[25] で表現した。URL とはインターネット上に存在する様々なリソースをあらわすためのアドレスであり、WWW で利用できる。

そこで、データベース接続情報を Z39.50 用 URL[25] で表現することで、現在の WWW の仕組みでデータベース接続情報を流通できるようになる。データベース接続情報を表す URL の書式は

z39.50s://ホスト名[:ポート番号]/データベース名

であり、例えば図書館情報大学デジタル図書館の opac データベースであれば次のように表される。

z39.50s://lib.ulis.ac.jp:210/opac

Z39.50 用 URL を用いてデータベース接続情報を加工したデータベースリストを記述した HTML ファイルを以下に示す。

```
<html>
<head>
<title>データベースリスト</title>
</head>
<body>
<h1>データベースリスト</h1>
<ul>
<li><a href="z39.50s://Tikal.dev.oclc.org:210/Medline">OCLC - First Search[Medline]</a>
<li><a href="z39.50s://Tikal.dev.oclc.org:210/MDX_Health">OCLC - First Search[MDX_Health]</a>
<li><a href="z39.50s://Tikal.dev.oclc.org:210/EnvironmentS">OCLC - First Search[EnvironmentS]</a>
</ul>
</body>
</html>
```

ここまでの過程つまり、データベース接続情報の収集・加工では、データベース接続情報を集めること、集めたデータベース接続情報の組織化を行った。ここではデータベース接続情報を RFC で定められた Z39.50 用 URL で記述することで、HTML のパイパーリンクで示すことができる。そのため、今までのように各データベース接続情報収集サイトやクライアントの記述を理解してデータベース接続情報を抜き出さなくてよく、本研究で開発した WWW ブラウザ（後の 2.6 節で説明）を利用して、リンクをたどるだけでよい。

また、データベース接続情報を Z39.50 用 URL であらわしたデータベースを、HTML で記述し WWW サーバ上に置くことで、この HTML ファイルの URL を指定すれば、インターネット上のどこからでも閲覧できる。

このようにデータベース接続情報を WWW で流通させることで、自分以外が作成したデータベースリストを、自分が作成したのと同様に利用できることで、個々の人が完全な収集や加工をしなくても良くなる。

2.6 データベース接続情報の利用

データベース接続情報の利用とは、WWW 上にある HTML で記述されたデータベース接続情報を表示したり、データベースに接続したり、検索したりすることである。

データベースを検索するには、データベース接続情報を表示したり、データベース接続情報を登録して検索できるアプリケーションが必要である。1 章でも述べたように、現在、WWW が爆発的に流行した背景には Mosaic という WWW を簡単に利用できるアプリケーションの登場がある。どんなに、すばらしい標準でも、実際に使うことのできるアプリケーションがなければ、データベースを検索することは出来ないからである。現在、データベース接続情報の流通・利用のために必要な Z39.50 用 URL のリンク情報を正しく解釈できる WWW ブラウザと日本語と、Z39.50 用 URL をデータベース接続情報として解釈できる Z39.50 クライアントは存在しないため、Z39.50 用 URL を利用でき

る WWW ブラウザの開発と、この WWW ブラウザから直接起動できる Z39.50 クライアントの開発を行なった。

2.6.1 WWW ブラウザ

ユーザは本 WWW ブラウザを用いて WWW 上にある HTML ファイルをみてデータベース接続情報を収集したり、Z39.50 用 URL のリンクをたどることができる。

システム構成

WWW ブラウザは日本語化 Tcl/Tk8.0.5 で書かれたフリーで提供されている HTML ブラウザ [26] に、HTTP プロトコルの解析機能と、Z39.50 クライアントを呼び出す機能を付加して開発した。HTTP プロトコルの解析は Tcl/Tk 附属の HTTP ライブラリを利用した。

機能

- HTML 文書の表示
- HTML ファイルを HTTP を介して取得
- Z39.50 用 URL を解析し、Z39.50 クライアントを起動
- URL を直接指定して HTML ファイルを取得

2.6.2 Z39.50 クライアント [27][28][29]

ユーザは本検索クライアントを用いて著者や書名などのアクセスポイントを指定したり、前方一致や後方一致、履歴を利用して検索できる。また、レコード数、履歴集合、詳細度を指定して検索結果を見ることができる。

クライアントの役割は GUI を通したユーザからの接続要求、検索要求、返戻要求を解析・変換し、サーバに対して実際の要求を行うことである。それぞれの要求はプロトコルエンジンによって APDU に変換されてサーバに渡され、サーバから返された APDU は出した要求に応じてクライアント内で処理される [30]。

構成

Z39.50 クライアントは日本語化 Tcl/Tk8.0.5 で開発し、Z39.50 プロトコルの解析は Index Data 社の YAZ (Z39.50 ライブラリ) [31] を利用した。

機能

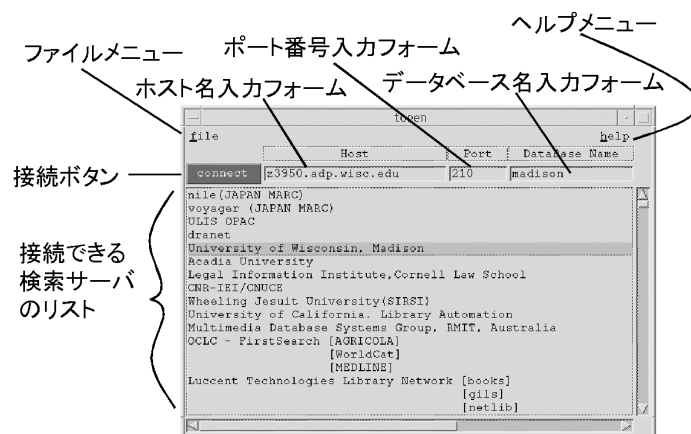


図 2.2: 接続ウィンドウの機能

図 2.2は接続ウィンドウの画面である。画面中央には OCLC First Search や University of California など約 200 のサーバのリストが表示されており、ユーザが接続したいサーバを選択して connect ボタンをクリックすると、選択したサーバの IP アドレス、ポート番号、データベース名が画面上部のウィンドウ内に入力され、接続が開始される。リストにないサーバに接続したい場合は、キーボードから直接サーバ情報をウィンドウに入力できる。また、サーバ情報を表す URL を指定ファイルに登録することによりリストにサーバ名を追加できる。さらに、現在は Index Data 社が作成しているサーバリスト [9] を定期的に参照することによりサーバリストを自動的に更新しており、エンドユーザは常に最新の情報に基づいてサーバを選択できる。

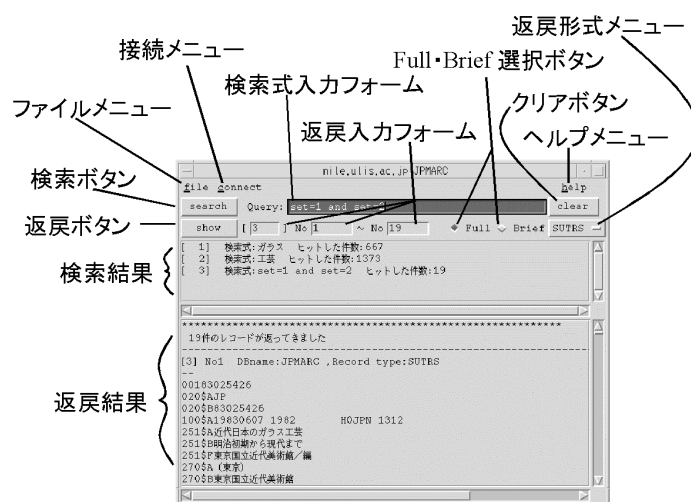


図 2.3: 検索ウィンドウの機能

図 2.3は検索ウインドウ画面である。ユーザは検索式の入力、検索ボタンのクリック、検索結果の確認、結果集合の指定、返戻ボタンのクリック、返戻結果の確認という一連の操作を検索ウインドウ上で行う。その際、入力フォームやボタンがユーザの操作に連動して色が変わり、ユーザが次にどの操作をすべきかがわかるようになっている。

ユーザが検索式を入力し、search ボタンをクリックすると、検索結果集合番号、検索式、ヒット件数が表示される。新しく検索式を入力したい場合には clear ボタンをクリックすればよい。検索結果の確認後、検索結果の内容を見たい場合は、表示させたい検索集合番号をクリックして show ボタンをクリックすると指定した検索集合のレコードが表示される。検索集合を指定せずに show ボタンをクリックすると、直前の検索結果集合のレコードが表示される。また、キーボードから検索結果集合番号、開始レコード番号、終了レコード番号の直接指定もできる。

検索式の書式は検索コマンドの標準規格 (ANSI Z39.58-1992) である CCL(Common Command Language) を採用した。これはもともと YAZ が CCL から逆ポーランド記法の検索式に変換する [32] 機能を持っているためである。返戻結果の表示は Brief、Full ボタンをクリックすることで、簡易表示 (Brief) ・詳細表示 (Full) を選択できる。また、返戻形式メニューから USMARC や SUTRS などのレコード形式を選択できる。

ファイルメニューの「New Window」を選択すると、現在接続しているサーバに接続したまま、新しい接続ウインドウを起動させ、他のサーバに接続できる。接続メニューの「Disconnect」を選択すると、現在接続しているサーバと接続を切り、新しい接続ウインドウが起動し、他のサーバに接続できる。ファイルメニューの「Exit」を選択するとクライアントを終了する。ヘルプメニューを選択すると、検索式の例が表示される。

2.7 システムの実行例

最初に、WWW ブラウザが起動する。初期画面を図 2.4 に示す。



図 2.4: WWW ブラウザ初期画面

初期画面の WWW ページの URL は `http://cosmo.ulis.ac.jp/~yuka/wkappa/index.html` である。この WWW ブラウザは既存の WWW ブラウザと同様に WWW 上のコンテンツであれば URL を入力エントリーにキーボードから直接入力したり、リンクをたどることで、表示できる。

図 2.5、図 2.6、図 2.7 は、今回収集・加工したデータベース接続情報をこの WWW ブラウザで表示したものである。

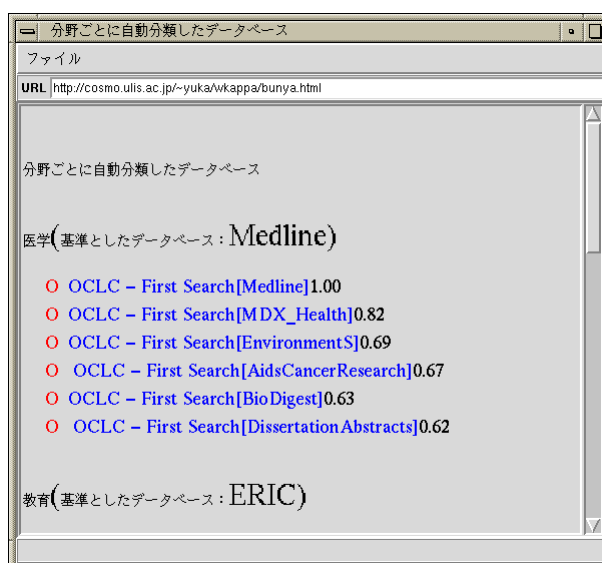


図 2.5: 分野毎に自動分類したデータベースリスト

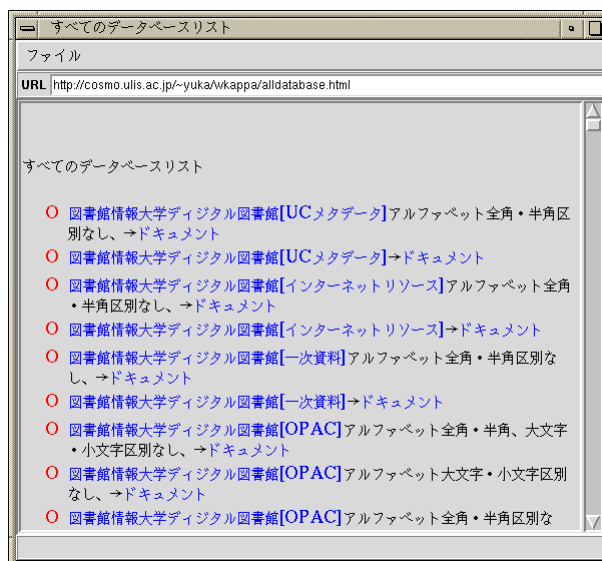


図 2.6: サーバごとに並び変えたデータベースリスト



図 2.7: 良く使うデータベースリスト

この WWW ブラウザは Z39.50 用 URL を解釈し、Z39.50 用の URL のリンクをたどれば、URL であらわされたデータベースに接続した Z39.50 クライアントを起動する。図 2.8 は図書館情報大学デジタル図書館のデータベースのリンクをたどり、データベースに接続した Z39.50 クライアントが起動したところである。

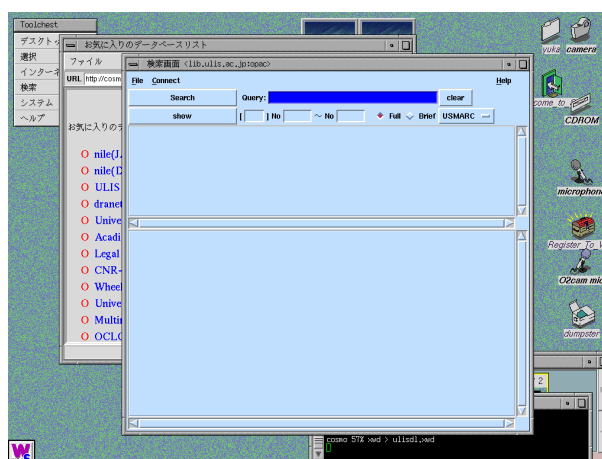


図 2.8: Z39.50 クライアントの起動画面

そして、Z39.50 クライアントで実際にデータベースを検索できる。図 2.9 は www で検索し、検索結果を表示したところである。



図 2.9: Z39.50 クライアントの検索画面

また、開発したシステムは、データベースに接続したまま、他のデータベースも接続できる。

図 2.10 は、図書館情報大学デジタル図書館のデータベースに接続したまま Japan/Marc のデータベースに接続し、同様に www で検索し検索結果を表示したところである。

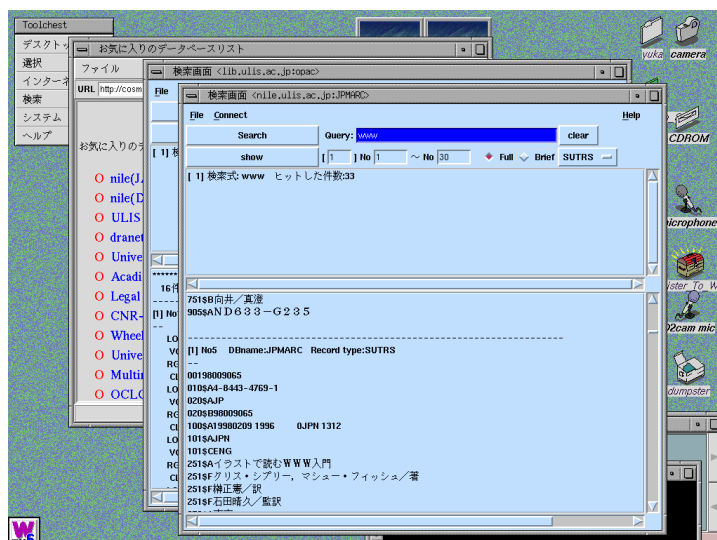


図 2.10: 複数データベース検索実行例

このように同時に複数のデータベースに接続できる。また、データベースに接続したまま状態で WWW ブラウザを利用してドキュメントを見ることもできる。図 2.11 は、データベースを検索しつつ、ヘルプドキュメントを表示したところである。

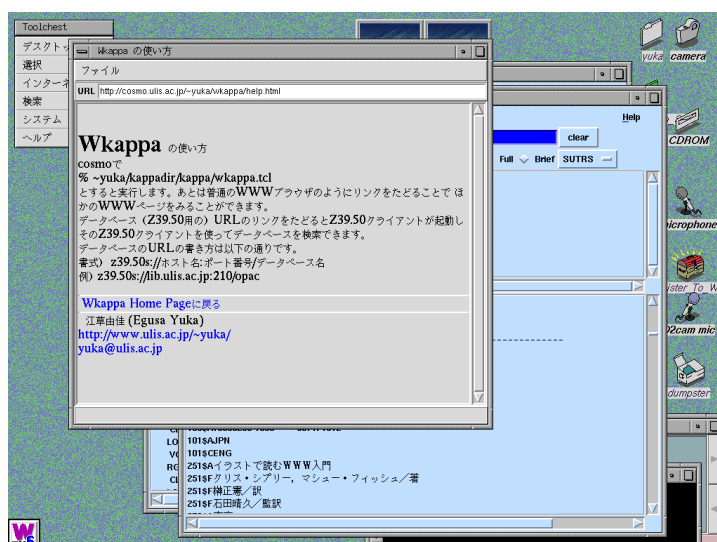


図 2.11: ドキュメント表示例

つぎに Z39.50 クライアント単独で実行した例をあげる。



図 2.12: 検索ウィンドウ : Japan/MARC サーバ

まず、Japan/MARC サーバを検索した例を図 2.12に示す。検索結果表示欄を元に検索を順に追うと、

1. 「ガラス」を検索 ヒットした件数が 667 件
2. 「工芸」を検索 ヒットした件数が 1373 件
3. 1 番目と 2 番目の検索結果集合の AND 検索 ヒットした件数が 19 件

という検索要求が出されていることがわかる。3 番目の検索式の “set=1” は 1 番目の検索集合を “set=2” は 2 番目の検索集合を指している。“and” はその AND をとるという意味である。返戻結果表示部分は 3 番目の検索結果集合を表示している。



図 2.13: 検索ウィンドウ : 図書館情報大学デジタル図書館システム Z39.50 サーバ

次に、Japan/MARC サーバ以外の日本語を用いたサーバを検索している例を図 2.13に示す。図 2.13は図書館情報

大学デジタル図書館システム Z39.50 サーバを検索しているところである。検索結果表示欄を元に検索を順に追うと、

1. 「統計」を検索 ヒットした件数が 1425 件
2. 「書名に統計」を検索 ヒットした件数が 1154 件
3. 「書名に統計」と「SPSS」の AND 検索 ヒットした件数が 5 件

という検索要求が出されていることがわかる。2 番目と 3 番目の検索式の“ti=” はアクセスポイントに書名を使うことを表している。3 番目の検索式の“and” は「書名に統計」と「SPSS」の AND をとるという意味である。返戻結果表示部分は 3 番目の検索結果集合を表示している。

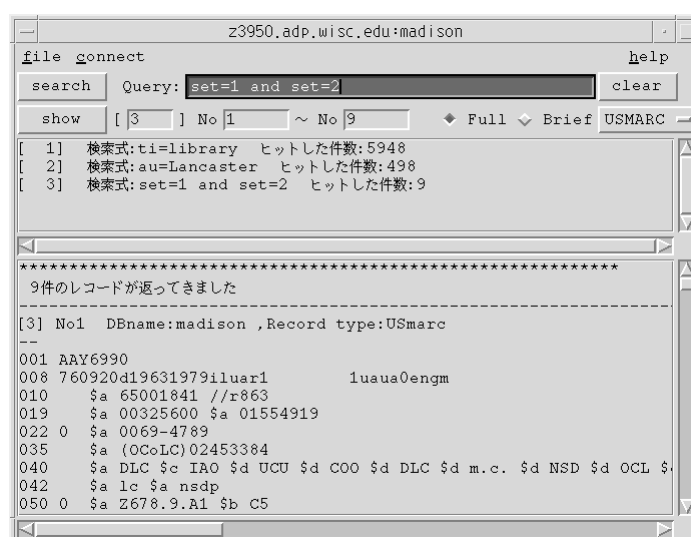


図 2.14: 検索ウインドウ : University of Wisconsin, Madison

最後に海外のサーバを検索しているところを図 2.14に示す。図 2.14は University of Wisconsin, Madison のサーバを検索しているところである。検索結果表示欄をもとに検索を順に追うと、

1. 「書名に library」を検索 ヒットした件数が 5948 件
2. 「著者名に Lancaster」を検索 ヒットした件数が 498 件
3. 1 番目と 2 番目の検索結果集合の AND をとる検索 ヒットした件数が 9 件

という検索要求が出されていることがわかる。1 番目の検索式の“ti=” はアクセスポイントに書名を使うことを表しており、2 番目の検索式の“au=” はアクセスポイントに著者名を使うことを表している。3 番目の検索式の“set=1” は 1 番目の検索集合を“set=2” は 2 番目の検索集合を指している。“and” はその AND をとるという意味である。返戻結果表示部分は 3 番目の検索結果集合を表示している。

第3章

考察

これまで述べてきたように、本研究ではデータベース接続情報の収集・加工・流通・利用のサイクルを一連の流れで利用できるシステムを開発することで、データベース選択支援機能を実現した。

この章では既存のデータベース選択支援に関連するシステムと本研究との違いについて考察する。また、本研究で対象とした領域について特に Z39.50 と WWW との関わりと今後について考察する。

3.1 既存のデータベース選択支援システムとの比較

WAIS(Wide Area Information Servers) のように本研究のアプローチとは異なるアプローチでデータベース選択支援機能を実現しているものも存在する。

WAIS のデータベース選択支援機能のアプローチは、データベースに関する情報を扱う特別なデータベース `directory-of-servers` を設け、利用者がこのデータベースを検索することで、目的のデータベースを見つけ、データベースに接続するための情報を知ることができる [33] ものである。しかし、利用者がデータベースを見つける手段は、`directory-of-servers` を検索するしかない。

WAIS のアプローチと本研究のアプローチではデータベース接続情報を利用者サイドから発信できるかどうか異なる。WAIS では `directory-of-servers` という特別なデータベースを作成しなければならない。このデータベースを作成し更新していくのは簡単ではない。しかし、本研究ではデータベース接続情報を記述した HTML を作成して WWW 上に置くだけで良く、現在の WWW コンテンツと同様に扱えるため、利用者の側からも容易にデータベース接続情報を発信できる。

3.2 Z39.50 と WWW との関わり

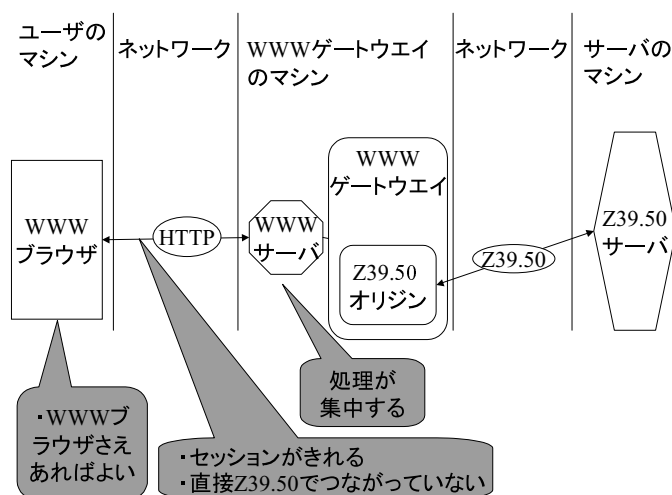


図 3.1: WWW-Z39.50 ゲートウェイシステムとプロトコルとの関係

既存のシステムには WWW-Z39.50 ゲートウェイシステムや ZAP[34] のように Z39.50 と WWW を組み合わせているものもある。(図 3.1参照)

WWW-Z39.50 ゲートウェイシステムは、現在、ほとんどのマシンにインストールされている WWW ブラウザのみでユーザは検索できるので、多くの機関で採用されている。

ゲートウェイシステムでは、ユーザからの検索要求は HTTP プロトコルを介して、いったん WWW サーバに送られ、WWW ゲートウェイが Z39.50 プロトコルに変換して Z39.50 データベースに接続する。また、Z39.50 データベースからの返信は、WWW ゲートウェイが Z39.50 を解析し、HTTP を介してユーザに送られる。

このシステムでは、HTTP プロトコルから Z39.50 プロトコルへの変換を行っているため、処理が複雑になるという問題がある。また、全てのユーザの検索処理は WWW サーバに集中してしまう、ユーザの細かな指定ができないという問題もある [35]。

現在の WWW の普及はめざましいものがあり、1999 年の 1 月時点で、WWW ページは 8 億あるとも言われている [36]。その普及の背景には、グラフィカルな WWW ブラウザの出現、HTTP プロトコルのシンプルさ、HTML ファイルを作成すればよいというユーザからの情報発信の容易さがあると筆者は考える。しかし、既存の Z39.50-WWW ゲートウェイでは、WWW から Z39.50 への変換という複雑な処理に加え、ユーザからデータベース接続情報を発信できない。

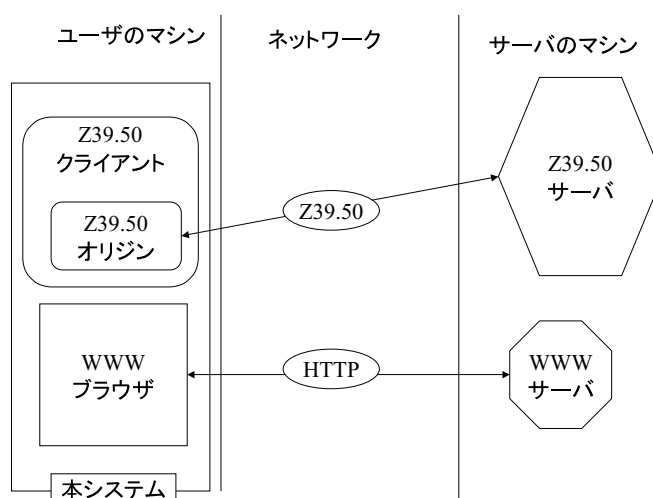


図 3.2: 本システムのプロトコルとの関係

そこで本研究のように、情報検索には Z39.50 を使い、情報共有には WWW をというように、Z39.50 と WWW のそれぞれに適した役割に使い分けて使用すること（図 3.2 参照）で、WWW システムのシンプルな構成を残したまま、WWW を利用し、検索に適した Z39.50 を利用し得ることで、WWW の普及と同様に Z39.50 の普及がすすみ、情報共有と情報検索の向上につながると筆者は考える。

第4章

おわりに

本研究では、検索利用者のデータベース選択支援を目的として、データベース接続情報の収集・加工・流通・利用を一連の流れで行えるシステムを開発した。今回の、データベース選択支援のアプローチはデータベース接続情報の収集・加工・流通・利用を一連の流れでできるようにすることにより、Z39.50 データベースを検索利用者が、データベースを選択するための手段を増やすことである。

データベース接続情報が流通できるようになったことで、Z39.50 データベース接続情報の利用者間での交換が可能になった。

ユーザがデータベース接続情報を発信するようになれば、現在、WWW で可能となっているロボットでのデータベース接続情報の収集も可能になる。

なお、本研究の一部は情報知識学会研究報告会 [28]、情報知識学会誌 [29]、デジタル図書館ワークショップ [37] で発表した。

謝辞

石塚英弘先生、宇陀則彦先生には研究全般にあたり指導していただきましてありがとうございました。毎週のゼミで様々な指摘をしてくれた石塚研ゼミのみなさん、特に斎藤ひとみさん、伊東千夏さんにはプレゼンテーションの練習において、高久雅生さんにはシステム構築にあたり数多くの助言をいただきました。本当にありがとうございました。

参考文献

- [1] Lynch, Clifford A. The Z39.50 Information Retrieval Standard : Part I: A Strategic View of Its Past, Present and Future. D-Lib Magazine. (1997.4).
URL: <<http://www.dlib.org/dlib/april97/04lynch.html>>.
- [2] 上田修一. Z39.50 とその可能性. 情報の科学と技術. Vol.48, No.3, p.126-133 (1998.3).
- [3] Comer, Douglass 著、村井純・楠本博之訳. “第1章イントロダクション及び概要”. TCP/IP によるネットワーク構築. 東京, トッパン, 1995, p.1-13.
- [4] 益岡竜介, 木庭袋圭祐. World-Wide Web. 情報処理. Vol.36, No.12, p.1155-1165 (1995.12).
- [5] 白橋明弘. マルチメディア時代のインターネット技術. インターフェイス. Vol. 21, No. 9, p.64-73 (1995.9).
- [6] 牛崎進. Z39.50—IR サービスの通信プロトコル—. カレントアウェアネス. No.175, p.4 (1994.3).
- [7] 上田修一. Z39.50 の可能性と問題点. 三田図書館・情報学会研究大会予稿集 (1996.11).
- [8] 松林正己. 標準情報検索プロトコル Z39.50 の国際的展望. 情報の科学と技術. Vol.48, No.3, p.144-155 (1998.3).
- [9] Index Data. Z39.50 Target Information. last update 2000-01-09, Continually updated.
URL: <<http://www.indexdata.dk/targettest/targetstat.shtml>>.
- [10] JIS X 0806 (ISO 23950). 情報検索 (Z39.50) 応用サービス定義及びプロトコル仕様. 東京, 日本規格協会, 1999, 272p.
- [11] 図書館情報大学附属図書館. 図書館情報大学デジタル図書館 (ULIS-DL). (参照 2000-1-9).
URL: <<http://lib.ulis.ac.jp/>>.
- [12] Titech Digital Library (東京工業大学電子図書館). last update 1999-3-25.
URL: <<http://tdl.libra.titech.ac.jp/>>.
- [13] 小島明, 田中純, 篠原正紀, 落合崇道. 大学での取り組みが進む電子図書館. NTT 技術ジャーナル. Vol.11, No.1, p.55-58 (1999.1).
- [14] 原田昌紀. “第1章 WWW とサーチエンジン”. サーチエンジン徹底活用術. 東京, オーム社, 1997, p.1-18.
- [15] ANSI/NISO Z39.50-1995. Information Retrieval (Z39.50) : Application Service Definition and Protocol Specification. 1995, 156p.
URL: <<ftp://ftp.loc.gov/pub/z3950/official/>>.
- [16] Hammer, Sebastian. Z39.50 and the World Wide Web. D-lib magazine. (1996.3).
- [17] BookWhere2000. (参照 2000-1-9).
URL: <<http://www.bookwhere.com/>>.
- [18] Willow Information Center. (参照 2000-1-9).
URL: <<http://www.washington.edu/willow/>>.
- [19] UKOLN. Directory of Z39.50 targets in the UK. last update 1998-7-17.

- URL: <<http://www.ukoln.ac.uk/dlis/zdir/>>.
- [20] NOVANET. Directory of Z39.50 Targets in Canada. (参照 2000-1-9).
URL: <<http://novanet.ns.ca/vCucadm.html>>.
- [21] National Library of Australia. Directory of Z39.50 Targets in Australia.last update 1998-10-23.
URL: <<http://enzo.nla.gov.au/products/alg/adm/vCucadm.html>>.
- [22] Cornell University Library. Cornell University Library Gateway. (参照 2000-1-9)
URL: <<http://campusgw.library.cornell.edu/>>.
- [23] Payette, Sandra D.; Rieger, Oya Y. Z39.50 : The User's Perspective. D-lib magazine. 1997.
URL: <<http://www.dlib.org/dlib/april97/cornell/04payette.html>>.
- [24] 杉山高一. “第 1 章 相関係数”. 多変量データ解析入門. 東京, 朝倉書店, 1983, p.1-12.
- [25] Denenberg, R.; Kunze, J.; Lynch, D. eds. RFC 2056 : Uniform Resource Locators for Z39.50. 1996.
URL: <<http://www.ietf.org/rfc/rfc2056.txt>>.
- [26] 須栗歩人. “4-12-6 text widget のその他の機能”. 入門 Tcl/TK. 東京, 株式会社秀和システム, 1998, p. 168-175.
(ISBN 4-87966-795-1)
URL: <<http://www.shuwasystem.co.jp/books/795-s.htm>>.
- [27] 江草由佳.Z39.50 プロトコルを用いた検索クライアントの開発. つくば, 図書館情報大学, 1998. 卒業研究論文.
- [28] 江草由佳, 真野泰久, 宇陀則彦, 石塚英弘. Z39.50 プロトコルによる日本語書誌データ情報検索システム. 第 6 回研究報告会講演論文集. 情報知識学会. 東京, 1998-5. 情報知識学会, 1998, p.29-36.
- [29] 宇陀則彦, 江草由佳, 高久雅生, 石塚英弘. Z39.50 による日本語書誌データ検索システム. 情報知識学会誌. Vol.9, No.2, p.1-15 (1999.9) .
- [30] 安齋宏幸. Z39.50 の技術解説. 情報の科学と技術. Vol.48, No.3, p.134-139 (1998.3) .
- [31] Index Data. The YAZ Toolkit. last update 1999-12-10.
URL: <<http://www.indexdata.dk/yaz/>>.
- [32] Index Data. YAZ User's Guide and Reference: Introduction. (参照 1999-5-4) .
URL: <<http://www.indexdata.dk/yaz/yaz.shtml>>.
- [33] 阪口哲男. Internet における情報資源の利用と提供の実際. デジタル図書館. No. 1, p. 43-50 (1994.8) .
URL: <http://www.dl.ulis.ac.jp/DLjournal/No_1/saka/saka.html>.
- [34] Index Data. ZAP!.last updated 1999-12-10.
URL: <<http://www.indexdata.dk/zap/>>.
- [35] Turner, Fay. Selecting a Z39.50 Client Or Web Gateway. LIBRARY HI TECH. Vol.16, No. 2, p.7-18 (1998).
- [36] 原田昌紀. WWW ロボットとサーチエンジンのスケーラビリティ. bit. Vol. 31, No. 12, p.22-28 (1999.12) .
- [37] 高久雅生, 江草由佳, 宇陀則彦, 石塚英弘. Z39.50 による書誌データ検索システムの構築-Dublin Core を共通スキーマとして-. デジタル図書館. No.16, p.97-106 (1999.11) .

参考文献

- 特集：Z39.50. 情報の科学と技術. vol. 48, no. 3, p. 125-180 (1998.3).
- 図書館情報学ハンドブック編集委員会 編. “7. 情報検索システム”. 図書館情報学ハンドブック. 東京, 丸善, 1988, p. 593-701. (ISBN 4-621-03232-1).
- 松原望. わかりやすい統計学. 東京, 丸善株式会社, 1996, 149p. (ISBN 4-915666-27-1).

付録

付録として本研究で開発したシステムのソースプログラムと作成したデータベースリストを添付した。

付録 A

データベースリスト

A.1 お気に入りのデータベースリスト

```

<html>
<head>
<title>お気に入りのデータベースリスト</title>
</head>

<body>
<h1>お気に入りのデータベースリスト</h1>
<ul>
  <li> <a href="z39.50s://nile.ulis.ac.jp:9999/JPMARC">nile(JAPAN MARC
)</a>
  <li> <a href="z39.50s://nile.ulis.ac.jp:11111/dc">nile(DC)</a>
  <li> <a href="z39.50s://lib.ulis.ac.jp:210/opac">ULIS OPAC</a>
  <li> <a href="z39.50s://dranet.dra.com/drewdb">dranet</a>
  <li> <a href="z39.50s://z3950.adp.wisc.edu/madison">University of Wi
sconsin, Madison</a>
  <li> <a href="z39.50s://jasper.acadiau.ca:2200/UNICORN">Acadia Unive
rsity</a>
  <li> <a href="z39.50s://supct.law.cornell.edu/SYLLABI">Legal Informa
tion Institute,Cornell Law School</a>
  <li> <a href="z39.50s://arca.iei.pi.cnr.it:2000/IEI-Books">CNR-IEI/C
NUCE</a>
  <li> <a href="z39.50s://192.188.131.54:2200/UNICORN">Wheeling Jesuit
University(SIRSI)</a>
  <li> <a href="z39.50s://melvyl.ucop.edu/CAT">University of Californi
a. Library Automation</a>
  <li> <a href="z39.50s://z3950.mds.rmit.edu.au/ZIW">Multimedia Databa
se Systems Group, RMIT, Australia</a>
  <li> <a href="z39.50s://tikal.dev.oclc.org/AGRICOLA">OCLC - FirstSea
rch [AGRICOLA]</a>
  <li> <a href="z39.50s://tikal.dev.oclc.org/WorldCat">OCLC - FirstSea
rch [WorldCat]</a>
  <li> <a href="z39.50s://tikal.dev.oclc.org/MEDLINE">OCLC - FirstSea
rch [MEDLINE]</a>

```

```

<li> <a href="z39.50s://z3950.bell-labs.com/books">Luccent Technolog
ies Library Network [books]</a>
<li> <a href="z39.50s://z3950.bell-labs.com/gils">Luccent Technolog
ies Library Network[gils]</a>
<li> <a href="z39.50s://z3950.bell-labs.com/netlib">Luccent Technolo
gies Library Network[netlib]</a>
<li> <a href="z39.50s://z3950.bell-labs.com/factbook">Luccent Techno
logies Library Network[factbook]</a>
<li> <a href="z39.50s://z3950.ovid.com:2213/pmed">Ovid [pmed]</a>
<li> <a href="z39.50s://z3950.ovid.com:2213/eric">Ovid [eric]</a>
<li> <a href="z39.50s://z3950.ovid.com:2213/chid">Ovid [chid]</a>
</ul>

```

```
<hr>
```

```
<a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/index.html">Wkappa Home
Page に戻る</a>
```

```
<hr>
```

```
<address>
```

```
江草由佳 (Egusa Yuka)<br>
```

```
<a href="http://www.ulis.ac.jp/~yuka/">http://www.ulis.ac.jp/~yuka/</a>
<br>
```

```
<a href="mailto:yuka@ulis.ac.jp">yuka@ulis.ac.jp</a>
```

```
</address>
```

```
</body>
```

```
</html>
```

A.2 分野事に自動分類したデータベースリスト

```
<html>
```

```
<head>
```

```
<title>分野ごとに自動分類したデータベース</title>
```

```
</head>
```

```
<body>
```

```
<h1>分野ごとに自動分類したデータベース</h1>
```

```
<h2>医学 (基準としたデータベース : Medline)</h2>
```

```
<ul>
```

```
<li><a href="z39.50s://Tikal.dev.oclc.org:210/Medline">OCLC - First
Search[Medline]</a>1.00
```

```
<li><a href="z39.50s://Tikal.dev.oclc.org:210/MDX_Health">OCLC - Fir
st Search[MDX_Health]</a>0.82
```

```
<li><a href="z39.50s://Tikal.dev.oclc.org:210/EnvironmentS">OCLC - F
irst Search[EnvironmentS]</a>0.69
```

```
<li><a href="z39.50s://Tikal.dev.oclc.org:210/AidsCancerResearch">
OCLC - First Search[AidsCancerResearch]</a>0.67
```

OCLC - First Search[BioDigest]0.63

OCLC - First Search[DissertationAbstracts]0.62

<h2>教育 (基準としたデータベース : ERIC)</h2>

OCLC - First Search[ERIC]1.00

OCLC - First Search[DissertationAbstracts]0.72

OCLC - First Search[EducationIndex]0.61

<h2>経済 (基準としたデータベース : ABI-INFORM)</h2>

OCLC - First Search[ABI-INFORM]1.00

OCLC - First Search[WilsonBusinessAbstracts]0.87

OCLC - First Search[PeriodicalAbstracts]0.81

>

OCLC - First Search[ReadersGuideAbstracts]0.80

OCLC - First Search[BioDigest]0.76

OCLC - First Search[NewYorkTimes]0.67

OCLC - First Search[GEODATABASE]0.67

OCLC - First Search[FactSearch]0.67

OCLC - First Search[DissertationAbstracts]0.66

OCLC - First Search[MDX_Health]0.62

現在はこの3分野でしか自動分類は行っていません。

```

<hr>
<h1>ドキュメント・リンク</h1>
<ul>
  <li><a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/helpzclient.html">
Z39.50 クライアントの使い方</a>
</li>
</ul>

<hr>
<a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/index.html">Wkappa Home
Page に戻る</a>
<hr>
<address>
江草由佳 (Egusa Yuka)<br>
<a href="http://www.ulis.ac.jp/~yuka/">http://www.ulis.ac.jp/~yuka/</a>
<br>
<a href="mailto:yuka@ulis.ac.jp">yuka@ulis.ac.jp</a>
</address>
</body>
</html>

```

A.3 データベース名ごとに並び替えたデータベースリスト

```

<html>
<head>
<title>すべてのデータベースリスト</title>
</head>

<body>
<h1>すべてのデータベースリスト</h1>
<ul>
  <li><a href="cantata.ulis.ac.jp:210/ucmeta">図書館情報大学デジタル
図書館 [UC メタデータ]</a> アルファベット全角・半角区別なし、 <a href="h
ttp://lib.ulis.ac.jp/ulisd1/docs/z3950.html">ドキュメント</a>
  <li><a href="cantata.ulis.ac.jp:210/ucmeta-d">図書館情報大学デジタ
ル図書館 [UC メタデータ]</a> <a href="http://lib.ulis.ac.jp/ulisd1/docs
/z3950.html">ドキュメント</a>
  <li><a href="cantata.ulis.ac.jp:210/www">図書館情報大学デジタル図
書館 [インターネットリソース]</a> アルファベット全角・半角区別なし、 <a
href="http://lib.ulis.ac.jp/ulisd1/docs/z3950.html">ドキュメント</a>
  <li><a href="cantata.ulis.ac.jp:210/www-d">図書館情報大学デジタル
淵銭 [インターネットリソース]</a> <a href="http://lib.ulis.ac.jp/ulis
dl/docs/z3950.html">ドキュメント</a>
  <li><a href="cantata.ulis.ac.jp:210/prim">図書館情報大学デジタル図
書館 [一次資料]</a> アルファベット全角・半角区別なし、 <a href="http://
lib.ulis.ac.jp/ulisd1/docs/z3950.html">ドキュメント</a>

```

図書館情報大学デジタル
 図書館 [一次資料] <a href="http://lib.ulis.ac.jp/ulisd/docs/z3950
 .html">ドキュメント

図書館情報大学デジタル図
 書館 [OPAC]アルファベット全角・半角、大文字・小文字区別なし、 <a h
 ref="http://lib.ulis.ac.jp/ulisd/docs/z3950.html">ドキュメント

図書館情報大学デジタル
 図書館 [OPAC]アルファベット大文字・小文字区別なし、 <a href="http:
 //lib.ulis.ac.jp/ulisd/docs/z3950.html">ドキュメント

図書館情報大学デジタル
 図書館 [OPAC]アルファベット全角・半角区別なし、 <a href="http://lib
 .ulis.ac.jp/ulisd/docs/z3950.html">ドキュメント

図書館情報大学デジタ
 ル図書館 [OPAC]アルファベット全角・半角、大文字・小文字区別あり、 <
 a href="http://lib.ulis.ac.jp/ulisd/docs/z3950.html">ドキュメント

東京工
 業大学附属図書館電子図書館 [国際会議録データベース] <a href="http://
 tdl.libra.titech.ac.jp/dlib/documents/z39.50/Zproc.html">ドキュメン
 ト

東京工
 業大学附属図書館電子図書館 [テクニカルペーパーデータベース] <a href=
 "http://tdl.libra.titech.ac.jp/dlib/documents/z39.50/Zpaper.html">ド
 網瓮鷗

東京工業
 大学附属図書館電子図書館 [外国雑誌目次データベース]東工大学内のみサ
 ービス、 <a href="http://tdl.libra.titech.ac.jp/dlib/documents/z39.50
 /Ztoc.html">ドキュメント

東京
 工業大学附属図書館電子図書館 [東京工業大学紀要データベース]
 足 網瓮鷗

東京工業
 大学附属図書館電子図書館 [EES(Elsevier Electronic Subscriptions)]東
 工大学内のみ、 <a href="http://tdl.libra.titech.ac.jp/dlib/documents/
 z39.50/Zees.html">ドキュメント

東京工
 業大学附属図書館電子図書館 [東京工業大学図書館目録データベース]
 ドキュメント

東
 京工業大学附属図書館電子図書館 [理工学系ネットワークリソースデータベー
 ス] <a href="http://tdl.libra.titech.ac.jp/dlib/documents/z39.50/Zr
 esource.html">ドキュメント

東京
 工業大学附属図書館電子図書館 [理工学系サーチエンジン] <a href="http
 ://tdl.libra.titech.ac.jp/dlib/documents/z39.50/Zengine.html">ドキュ

瓮鷗

- No.1 Aberdeen University [dynix]
- No.2 Acadia U. [UNICORN]
- No.3 AGCI Elements of Change [eoc]
- No.4 Amarillo College [MARION]
- No.5 Ames P. L. [ZSERVER]
- No.6 Anne Arundel Cty. P. L. [MARION]
- No.7 AURIGA - CeDoc (biblioteche della provincia di Modena) [default]
- No.8 Australian National University [INNOPAC]
- No.9 B.C. Electronic Library Network [ELN-Serials]
- No.10 Bell Laboratories Library Network [books]
- No.11 Bell Laboratories Library Network [factbook]
- No.12 Bell Laboratories Library Network [gils]
- No.13 Bell Laboratories Library Network [netlib]
- No.14 Biblioteca de Castilla y Leon [AbsysBCL]
- No.15 Biblioteca de Castilla y Leon [AbsysCCFL]
- No.16 Bibliothèque Nationale du Québec [IRIS]
- No.17 Bibliothèque de l'Université Moncton [ADVANCE]
- No.18 Bob Strunz [rte]
- No.19 Boston U. [INNOPAC]
- No.20 Brock U. [INNOPAC]
- No.21 Brunel U. [UNICORN]
- No.22 California State U. Lib., Dominguez Hills [MAIN*BIBMAST]
- No.23 California

State U. Lib., Hayward [Innopac]
 No.24 Calif
 ornia State U. Lib., Humboldt [MAIN*BIBMAST]
 No.25 California
 State U. Lib., Los Angeles [Innopac]
 No.26 Californ
 ia State U. Lib., Monterey Bay [Voyager]
 No.27 Californ
 ia State U. Lib., Northridge [MAIN*BIBMAST]
 No.28 California S
 tate U. Lib., Pomona [Innopac]
 No.29 California
 State U. Lib., San Francisco [Innopac]
 No.30 California S
 tate U. Lib., San Luis Obispo [Innopac]
 No.31 California Sta
 te U. Lib., San Marcos [Innopac]
 No.32 California
 State U. Lib., Sonoma [Marion]
 No.33 Californ
 ia State U. Lib., Stanislaus [MAIN*BIBMAST]
 No.34 CAVAL [dyni
 x]
 No.35
 Cedar Rapids P.L. [cedrap]
 No.36 Center for R
 esearch Libraries [INNOPAC]
 No.37 Central Col
 lege [INNOPAC]
 No.38 Central North C
 arolina Library Consortium [INNOPAC]
 No.39 Centro
 de Informacion Biblioteca ITESM Campus Monterrey [innopac]
 No.40 Chinese Uni
 versity of Hong Kong [INNOPAC]
 No.41 CIE
 SIN/ATSDR Catalog [atsdr]
 No.42 CISTI [I
 NNOPAC]
 No.43 Cleveland Publi
 c Library [CPL]
 No.44 Colle
 ge o/t Holy Cross [MARION]
 No.45 Colorado Alliance
 of Research Libraries [AIM]
 No.46 Colorado Alliance

of Research Libraries [ARA]
 No.47 Colorado Alliance
 of Research Libraries [BEM]
 No.48 Colorado Alliance
 of Research Libraries [CDE]
 No.49 Colorado Alliance
 of Research Libraries [CSP]
 No.50 Colorado Alliance
 of Research Libraries [DPL]
 No.51 Colorado Alliance
 of Research Libraries [EST]
 No.52 Colorado Alliance
 of Research Libraries [FTM]
 No.53 Colorado Alliance
 of Research Libraries [LAM]
 No.54 Colorado Alliance
 of Research Libraries [LUT]
 No.55 Colorado Alliance
 of Research Libraries [MCC]
 No.56 Colorado Alliance
 of Research Libraries [MIN]
 No.57 Colorado Alliance
 of Research Libraries [NEL]
 No.58 Colorado Alliance
 of Research Libraries [NEU]
 No.59 Colorado Alliance
 of Research Libraries [NJC]
 No.60 Colorado Alliance
 of Research Libraries [OTE]
 No.61 Colorado Alliance
 of Research Libraries [PPC]
 No.62 Colorado Alliance
 of Research Libraries [PUE]
 No.63 Colorado Alliance
 of Research Libraries [RGU]
 No.64 Colorado Alliance
 of Research Libraries [RRC]
 No.65 Colorado Alliance
 of Research Libraries [STR]
 No.66 Colorado Alliance
 of Research Libraries [TKO]
 No.67 Colorado Alliance
 of Research Libraries [TRI]
 No.68 Colorado Alliance
 of Research Libraries [WYO]
 No.69 Common

wealth Scientific and Industrial Research Organisation [LIBR]
 No.70 Connecticut State University [ERIC]
 No.71 Connecticut State University [INNOPAC]
 No.72 COPAC [copac]
 No.73 COPAC [COPAC]
 No.74 COWLNET Consortium [INNOPAC]
 No.75 Curtin University of Technology [marion]
 No.76 DanBib Test Server [danbibv2]
 No.77 Danish Union Catalogue [danbibv2]
 No.78 Data Research Associates [DREWDB]
 No.79 Deakin University [INNOPAC]
 No.80 Dept of Foreign Affairs & International Trade [INNOPAC]
 No.81 Duke U. [MARION]
 No.82 Earth Resources Observation Systems [GLISNODE]
 No.83 Eastern Sierra Geospatial Data Clearinghouse [WMRSFGDC]
 No.84 École Polytechnique de Montréal Bibliothèque [ADVANCE]
 No.85 École Polytechnique de Montréal [ADVANCE]
 No.86 Emory Library Catalog [unicorn]
 No.87 ESTO (Japan) Asian/Pacific Catalog [esto]
 No.88 Five Colleges Database [INNOPAC]
 No.89 Flinders University Library [ZSERVER]
 No.90 Florida State Library [MARION]
 No.91 George Mason University Libraries [VOYAGER]
 No.92 Getty

Research Institute [INNOPAC]
 No.93 Glasgow Ca
 ledonian University [Dynix]
 No.94 Gla
 sgow University [INNOPAC]
 No.95 Global C
 hange Master Directory (GCMD) [prod]
 No.96 Griff
 ith University [ADVANCE]
 No.97 Grimes
 Public Library--Grimes, Iowa [main]
 No.98 groucho.
 lib.rochester.edu [voyager]
 No.99 Harrington Li
 brary Consortium [MARION]
 No.100 Herning Central
 bibliotek [S]
 No.101 Hong
 Kong Baptist U. [INNOPAC]
 No.102 Hong Kong
 University of Science & Technology [INNOPAC]
 No.103 ICCU [nopac]<
 /a>
 No.104 Index Da
 ta Test Target [gils]
 No.105 Index Da
 ta Test Target [marc]
 No.106 Indi
 ana U. [LUIN]
 No.107 Innovative
 Interfaces VEGA [INNOPAC]
 No.108 I
 nstitute of Child Health and Institute of Neurology, University Colleg
 e London [UNICORN]
 No.109 IN
 TEC Systems Laboratory Inc. 2 [usmarc]
 No.110 INTEC
 SYSTEMS LABORATORY INC. [isl]
 No.111 IN
 TEC Systems Laboratory Inc. [jpmarc]
 No.112 INTEC
 SYSTEMS LABORATORY INC. [rfc]
 No.113 Italian Nation
 al Library Service [nopac]
 No.114 Kansa
 s City P. L. [MARION]

 No.115 King's Coll
 ege London [KCL01]

 No.116 Leed
 s Metropolitan University [LIBERTAS]

 No.117 Leeds Un
 iversity [INNOPAC]

 No.118 Lewis & Cl
 ark College [INNOPAC]

 No.119 LI
 BIS-Net [xxdefault]

 No.120 Librar
 y of Congress [APIF]

 No.121 Librar
 y of Congress [AUTH]

 No.122 Library
 of Congress [BIB]

 No.123 Li
 brary of Congress [BKSFILES]

 No.124 Libra
 ry of Congress [BOOKS]

 No.125 Library
 of Congress [CF]

 No.126 Lib
 rary of Congress [JACKPHY]

 No.127 Librar
 y of Congress [MAPS]

 No.128 Library
 of Congress [MSS]

 No.129 Libra
 ry of Congress [MUSIC]

 No.130 Libra
 ry of Congress [NAMES]

 No.131 Library
 of Congress [NUC]

 No.132 Librar
 y of Congress [OCAT]

 No.133 Librar
 y of Congress [PERM]

 No.134 Lib
 rary of Congress [PREMARC]

 No.135 Lib
 rary of Congress [SERIALS]

 No.136 Li
 brary of Congress [SUBJECTS]

 No.137 Library

of Congress [VM]

 No.138 LIBRIS [libr]

 No.139 Linda Hall Library [davinci]

 No.140 LISWA (Western Australia) [INNOPAC]

 No.141 Louisiana Online University I.S. [ISAAC]

 No.142 Louisiana Online University I.S. [LINUS]

 No.143 Louisiana Online University I.S. [LOIS]

 No.144 Louisiana Online University I.S. [MAVEN]

 No.145 Louisiana Online University I.S. [NEON]

 No.146 Louisiana Online University I.S. [SIDNEI]

 No.147 Louisiana Online University I.S. [TECH]

 No.148 Louisiana State Library [ILLSearch]

 No.149 Maharishi University of Management [MARION]

 No.150 M APLIN [MAIN*BIBMAST]

 No.151 Marin County P. L. [INNOPAC]

 No.152 Mary Institute/Country Day School [MARION]

 No.153 Massachusetts Institute of Technology [advance]

 No.154 Merrimack Valley Library Consortium [cl_default]

 No.155 Metro Boston Library Network [MARION]

 No.156 Middlebury College [MIDCAT]

 No.157 Mississippi State U. [XXDEFAULT]

 No.158 Missouri Southern State College [LION]

 No.159 MIT/Mass GIS Digital Orthophoto Project [ORTHOFGDC]

 No.160 MnSCU/PAL

S [pals]

- No.161
Monash University [VOYAGER]
- No.162 Murdoch
University [Innopac]
- No.163 Napie
r University Library [Dynix]
- No.164 National Lib
rary of Australia [NLA]
- No.165 Nati
onal Library of Medicine -- History of Medicine Collection [DEFAULT]</
a>
- No.166 National
Library of Medicine [MED66]
- No.167 National
Library of Medicine [MED75]
- No.168 National
Library of Medicine [MED80]
- No.169 National
Library of Medicine [MED85]
- No.170 National
Library of Medicine [MED93]
- No.171 Nation
al Library of Medicine [MEDLINE]
- No.172 Nat
ional Library of Medicine [PREMEDLINE]
- No.173 National
Library of New Zealand [pinz]
- No.174 Natural R
esources Canada. [INNOPAC]
- No.175 Nebraska Geos
patial Data Clearinghouse [fgdc]
- No.176 New York S
tate Dept of Health [innopac]
- No.177 NOAA National
Data Center [FDL]
- No.178 NOAA National
Data Center [JST]
- No.179 NOAA Nationa
l Data Center [NCDC]
- No.180 NOAA/Cos
tal Services Center Library [UNICORN]
- No.181 Nordic Web Inde
x, Greenland/Faroe Islands [fogl]
- No.182 Nordic We
b Index, Island [Default]

 No.183 Nordic Web Index, Norway [nwi]

 No.184 North Carolina Coastal Library Consortium [INNOPAC]

 No.185 North Carolina State University [MARION]

 No.186 North of Boston Library Exchange - College [cl_default]

 No.187 North of Boston Library Exchange - Public [cl_default]

 No.188 North Texas GIS Consortium [fgdcmeta]

 No.189 Nottingham Trent University Library & Information Services [ADVANCE]

 No.190 Nova Scotia Provincial Library [AVR]

 No.191 Nova Scotia Provincial Library [CBR]

 No.192 Nova Scotia Provincial Library [CEH]

 No.193 Nova Scotia Provincial Library [CUR]

 No.194 Nova Scotia Provincial Library [ECR]

 No.195 Nova Scotia Provincial Library [NSP]

 No.196 Nova Scotia Provincial Library [PAR]

 No.197 Nova Scotia Provincial Library [SSR]

 No.198 Nova Scotia Provincial Library [WCR]

 No.199 Novanet Inc [ADVANCE]

 No.200 OCLC - First Search [ABI-INFORM]

 No.201 OCLC - First Search [Agricola]

 No.202 OCLC - First Search [AidsCancerResearch]

 No.203 OCLC - First Search [AidsCancerSubscription]

 No.204 OCLC - First Search [AppliedScienceIndex]

 No.205 OCLC - First Search [ArticleFirst]

 No.206 OCLC
- First Search [ArtIndex]

 No.207
OCLC - First Search [Arts_Humanities]

 No.208
OCLC - First Search [ATLAReligion]

 No.209 OCLC
C - First Search [BioDigest]

 No.210
OCLC - First Search [BiographyIndex]

 No.211
OCLC - First Search [BiolAgrIndex]

 No.212
OCLC - First Search [BookReviewDigest]

 No.213
OCLC - First Search [BooksInPrint]

 No.214
OCLC - First Search [BusinessOrgs]

 No.215 OCLC -
First Search [CINAHL]

 No.216
OCLC - First Search [ConsumersIndex]

 No.217
OCLC - First Search [ContentsFirst]

No.218 OCLC - First Search [DissertationAbstracts]

 No.219 OCLC
- First Search [Econlit]

 No.220
OCLC - First Search [EducationIndex]

 No.221 O
CLC - First Search [Environment]

 No.222
OCLC - First Search [EnvironmentS]

 No.223 OCLC - F
irst Search [ERIC]

 No.224
OCLC - First Search [EthnicNews95]

 No.225
OCLC - First Search [EthnicNewsWatch]

 No.226 OCL
C - First Search [EventLine]

 No.227 OC
LC - First Search [FactSearch]

 No.228 O
CLC - First Search [FactsOnFile]

 No.229 OCLC
- First Search [FastDoc]

 No
.230 OCLC - First Search [GeneralScienceIndex]

 No.231 OCLC
- First Search [GEODATABASE]

 No.232 OCLC -
First Search [GeoRef]

 No.
233 OCLC - First Search [GeoRefSubscription]

 No.2
34 OCLC - First Search [GPOMonthlyCatalog]

 No.235
OCLC - First Search [HumanitiesIndex]

 No.236 OCLC -
First Search [INSPEC]

 No.23
7 OCLC - First Search [LegalPeriodicals]

 No.2
38 OCLC - First Search [LibraryLiterature]

 No.239
OCLC - First Search [MarcWorldCat]

 No.240 OC
LC - First Search [MDX_Health]

 No.241 OCLC
- First Search [Medline]

 No.242
OCLC - First Search [MicrocompAbs]

 No.243
OCLC - First Search [MLABibliography]

 No.244 OCLC
- First Search [NetFirst]

 No.
245 OCLC - First Search [NewspaperAbstracts]

 No.246
OCLC - First Search [NewYorkTimes]

 No.247 OC
LC - First Search [PAISDecade]

 No.248 O
CLC - First Search [PapersFirst]

 No
.249 OCLC - First Search [PeriodicalAbstracts]

 No.25
0 OCLC - First Search [ProceedingsFirst]

 No.251 OCL
C - First Search [PsycFIRST]

 No.252 OCLC
 - First Search [Psycinfo]

 No.253 OCLC - First Search [ReadersGuideAbstracts]

 No.254 OCLC - F
 irst Search [RILM]

 No
 .255 OCLC - First Search [SocialSciencesIndex]

 No.256 OCLC
 - First Search [SocioAbs]

 No.257 OCLC - First Search [SociologicalAbstracts]

 No.258 OC
 LC - First Search [UnionLists]

<a href="z39.50s://Tikal.dev.oclc.org:210/WilsonBusinessAbstracts"
 > No.259 OCLC - First Search [WilsonBusinessAbstracts]

 No.260 OCLC
 - First Search [WorldCat]

 No.261 Ohio State
 University [INNOPAC]

 No.262 Old Co
 lony Library Network (OCLN) [xxdefault]

 No.263 Oxford
 University, U.K. [ADVANCE]

 No.264 Pasadena Pu
 blic Library [ADVANCE]

 No.265 Pennsylv
 a State U. [catalog]

 No.266 Pers
 Record Collection [Default]

 No.267 PROmemoria [c
 ini]

 No.268 PROmemoria
 [civici]

 No.269 PROmem
 oria [guggenheim]

 No.270 PROmem
 oria [querini_oa]

 No.271 PROmemo
 ria [querini_s]

 No.272 Que
 en Margaret University College [unicorn]

 No.273 Queen's
 University [QUCAARCH]

 No.274 Radford U
 . [INNOPAC]

 No.275 Research Libraries Group [SAM]

 No.276 Research Planning, Inc. [meta]

 No.277 River Bend Library System [cl_default]

 No.278 RMIT University [Advance]

 No.279 Sioux City P. L. [ZSERVER]

 No.280 Spokane Public Library [MARION]

 No.281 Spring Arbor College [UNICORN]

 No.282 Spring Hill College--Mobile, Alabama [UNICORN]

 No.283 State Library of Victoria, Australia [dynix]

 No.284 Steiermrkischen Landesbibliothek [miopac]

 No.285 Stirling University [dynix]

 No.286 Stonehill College [INNOPAC]

 No.287 SUN Y at Albany [MAIN*BIBMAST]

 No.288 Texas A & M [AMTAM]

 No.289 Texas A & M [GATAM]

 No.290 Texas A & M [IUTAM]

 No.291 Texas A & M [MCTAM]

 No.292 Texas A & M [PVTAM]

 No.293 Texas State Library [cl_default]

 No.294 The Danish Veterinary and Agricultural Library [Agroline]

 No.295 The Danish Veterinary and Agricultural Library [eco01]

 No.296 The Danish Veterinary and Agricultural Library [kvl01]

 No.297 The Danish Veterinary and Agricultural Library [lit01]

 No.298 TPU [BOOK
]

 No.299 Trinity C
 ollege Dublin [advance]

 No.300 U Laval [
 DIAP0]

 No.301 U Laval [
 DIDAC]

 No.302 U Laval [U
 LAV]

 No.303 U of B
 ritish Columbia [MARION]

 No.304 U of
 Calgary [UNICORN]

 No.305 U of Cali
 fornia - San Francisco [INNOPAC]

 No.306 U of Californ
 ia MELVYL [PE]

 No.307 U of
 Hull [INNOPAC]

 No.308 U of Ill
 inois [UICOPAC1]

 No.309 U of Ill
 inois [UICOPAC2]

 No.310 U of M
 anitoba [MARION]

 No.311 U of Massa
 chusetts at Amherst [INNOPAC]

 No.312 U of Michig
 an, Dearborn [INNOPAC]

 No.313 U of Mic
 higan [FLNT]

 No.314 U of Mic
 higan [MCAT]

 No.315 U of
 Minnesota [LUMINA]

 No.316 U of New B
 runswick [unicorn]

 No.317 U of N
 ewcastle upon Tyne [ADVANCE]

 No.318 U of N
 orth Carolina - Chapel Hill [MARION]

 No.319 U of Norther
 n Wales [INNOPAC]

 No.320 U of Penn
 sylvania Law Library [INNOPAC]

 No.321 U of Saska
 tchewan [INNOPAC]

 No.322 U of South
 Carolina [USCAN]

 No.323 U of Sout
 hern Alabama [LUMI]

 No.324 U of Stra
 thclyde [HORIZON]

 No.325 U o
 f Wales, Bangor [INNOPAC]

 No.326 U of Wisco
 nsin, Green Bay [UWGB]

 No.327 U
 of Wisconsin, Madison [madison]

 No.328 U of Wisco
 nsin, Whitewater [UWPCAT]

 No.329 U of Wollon
 gong [INNOPAC]

 No.330 U. of Alber
 ta [Marion]

 No.331 U. of Ark
 ansas [Marion]

 No.332 U. of Conne
 cticut Helth Center Lib. [UWWOPac]

 No.333 U. of Kentuck
 y [LUKY]

 No.334 UNI
 LINC - Sydney [MARION]

 No.335 Unit
 ed States Geological Survey [pubs]

 No.336 Universit
 y College Cork [INNOPAC]

 No.337 University C
 ollege Galway [Darcy]

 No.338 Uni
 versity of Alabama in Birmingham [VOYAGER]

 No.339 Univers
 ity of Crete - Rethymnon Library [logios]

 No.340 U
 niversity of Crete - Rethymnon Library [ptolemeos_ii]

 No.341 U
 niversity of Crete-Heraklio Library [ptolemeos_ii]

 No.342 Univer
 sity of East Anglia Library [default]

 No.343 Un
 iversity of Hull [INNOPAC]

- No.344 University of Manitoba [MARION]
- No.345 University of Northern British Columbia [MARION]
- No.346 University of Patras [ADVANCE]
- No.347 University of Scranton [MARION]
- No.348 University of Tennessee Knoxville [horizon]
- No.349 University of Western Australia [innopac]
- No.350 University of Western Australia [INNOPAC]
- No.351 University of Wisconsin, Parkside [UWPCAT]
- No.352 University of York [DYNIX]
- No.353 Universitde Montr 蛛1 [ADVANCE]
- No.354 Universitde Sherbrooke [SIBUS]
- No.355 Vanderbilt University [ACORN]
- No.356 Vigo County P. L. [MARION]
- No.357 Ville de Montr 蛛1 [Z3950S]
- No.358 Vrije Universiteit [cl_default]
- No.359 WAI S-based mail archive [Default]
- No.360 Wellesley College [INNOPAC]
- No.361 Western North Carolina Library Network [INNOPAC]
- No.362 Woods Hole Oceanographic Institution Library [VOYAGER]
- No.363 Wyoming Natural Resources Data Clearinghouse [WNRDC]
- No.364 Yale University [YALEOPAC]

<hr>

Wkappa Home

Page に戻る

```

<hr>
<address>
江草由佳 (Egusa Yuka)<br>
<a href="http://www.ulis.ac.jp/~yuka/">http://www.ulis.ac.jp/~yuka/</a>
><br>
<a href="mailto:yuka@ulis.ac.jp">yuka@ulis.ac.jp</a>
</address>
</body>
</html>
<html>
<head>
<title>Wkappa Home Page</title>
</head>

<body>
<h1>Z39.50 データベースリスト </h1>
<ul>
  <li><a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/bookmark.html">お気に入りのデータベース</a>
  <li><a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/bunya.html">分野ごとに自動分類したデータベー
ス</a>
  <li><a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/alldatabase.html">すべてのデータベース</a>
</ul>
<h1>WWW Page Link</h1>
<ul>
  <li><a href="http://cosmo.ulis.ac.jp/~yuka/wkappa/help.html">Wkappa の使い方</a>
  <li><a href="http://www.ulis.ac.jp/~yuka/">Wkappa 制作者 (江草由佳) のページ</a>
</ul>
<hr>
<address>
江草由佳 (Egusa Yuka)<br>
<a href="http://www.ulis.ac.jp/~yuka/">http://www.ulis.ac.jp/~yuka/</a><br>
<a href="mailto:yuka@ulis.ac.jp">yuka@ulis.ac.jp</a>
</address>
</body>
</html>
<html>
<head>
<title>Wkappa の使い方</title>
</head>

<body>
<h1>Wkappa の使い方</h1>
cosmo で<br>
% ~yuka/kappadir/kappa/wkappa.tcl<br>

```

とすると実行します。あとは普通の WWW ブラウザのようにリンクをたどることで

ほかの WWW ページをみることができます。

データベース (Z39.50 用の) URL のリンクをたどると Z39.50 クライアントが起動し

その Z39.50 クライアントを使ってデータベースを検索できます。

データベースの URL の書き方は以下の通りです。

書式) z39.50s://ホスト名:ポート番号/データベース名

例) z39.50s://lib.ulis.ac.jp:210/opac

<hr>

Wkappa Home Page に戻る

<hr>

<address>

江草由佳 (Egusa Yuka)

http://www.ulis.ac.jp/~yuka/

yuka@ulis.ac.jp

</address>

</body>

</html>

付録 B

データベース接続情報を抽出するプログラム

B.1 C シェルプログラム コード名 : addNewServer

```
#!/bin/csh

set date='date +%Y%m%d'

# ターゲット情報を取り寄せる
echo ターゲット情報を取り寄せる
wget http://www.indexdata.dk/targettest/targetstat.shtml

# コード変換
echo コード変換
nkf -e targetstat.shtml > targetstat.shtmle
\mv targetstat.shtmle targetstat.shtml

# 取り寄せた HTML ファイルを解析用ファイル (lineanalysis.data) に変換
echo 取り寄せた HTML ファイルを解析用ファイル (lineanalysis.data) に変換
./lineanalysis.pl < targetstat.shtml > lineanalysis.data

# 解析用ファイルから scan 機能があるサーバを抜き出した解析用ファイルを作成
echo 解析用ファイルから scan 機能があるサーバを抜き出したファイルを作成
/usr/local/bin/grep scan lineanalysis.data > lineanalysisScan.data

# Scan 解析用ファイルから URL に変換した URL ファイル
# (ScanZURL.opensite,ScanZURLOnly.opensite )を作成
echo Scan 解析用ファイルから URL に変換した URL ファイル
echo (ScanZURL.opensite,ScanZURLOnly.opensite)を作成
awk -F'\t' \
' {print "z39.50s://" $5 ":" $6 "/" $3 "# No." $1 " " $2 "[" $3]";}' \
lineanalysisScan.data > ScanZURL.opensite
awk -F'\t' ' {print "z39.50s://" $5 ":" $6 "/" $3;}' \
lineanalysisScan.data > ScanZURLOnly.opensite

# 解析用ファイルから URL に変換した URL ファイル (ZURL.opensite) を作成
echo 解析用ファイルから URL に変換した URL ファイル (ZURL.opensite) を作成
```

```
awk -F'\t' '{print "z39.50s://" $5 ":" $6 "/" $3 "# No." $1 " " $2 \
[" $3"]";}' lineanalysis.data > ZURL.opensite
```

```
# 解析用ファイルから URL に変換した HTML ファイル (HTML.opensite) を作成
echo 解析用ファイルから URL に変換した HTML ファイル (HTML.opensite) を作成
awk -F'\t' \
'{print "<li><a href=\"z39.50s://" $5 ":" $6 "/" $3 "\"> No." $1 \
" " $2 [" $3 "]</a>";}' lineanalysis.data > HTML.opensite
```

```
# 今までのリストに、変更部分を追加
echo 今までのリストに、変更部分を追加
cat default.opensite ZURL.opensite > opensite
```

```
# ターゲット情報のファイルの名前を変換
echo 各ファイルの名前を変換
\mv targetstat.shtml "./old/targetstat.shtml.$date"
\cp lineanalysis.data "./old/lineanalysis.data.$date"
\cp lineanalysisScan.data "./old/lineanalysisScan.data.$date"
\cp ZURL.opensite "./old/ZURL.opensite.$date"
\cp ScanZURL.opensite "./old/ScanZURL.opensite.$date"
\cp ScanZURLonly.opensite "./old/ScanZURLonly.opensite.$date"
\cp default.opensite "./old/default.opensite.$date"
\cp opensite "./old/opensite.$date"
```

```
echo 終了
```

B.2 CSV ファイルに変換するプログラム コード名 : lineanalysis.pl

```
#!/usr/local/bin/perl -w

$flag=0;
$line = 1;
$content = "";

while(<>) {
    chop;
    if ( /class=no(\d)/ ) {
$flag = $1;
    } elsif ( $flag ) {
s/<a href(.*)>(.*</a>/$2/gi;
s/<font color=red>(.*</font>/$1/gi;
s/<br>//gi;
    if ( $flag == 1 ) {
        $content .= "$line\t";
        $line++;
    }
$content .= $_ . "\t";
    if ( $flag == 9 ) {
        $content .= "\n";
    }
$flag = 0;
    }
}
print $content;
```

付録 C

WWW ブラウザのプログラム

Wkappa のインストールには Tcl/Tk ライブラリと Tcl/Tk 用の HTML ライブラリが必要です。

Tcl/Tk ライブラリ (Tcl/Tk 8.05jp) 日本語化されているものをインストールしないと日本語の入力表示ができません。以下から入手できます。ftp://ftp.sra.co.jp/pub/lang/tcl/

Tcl/Tk 用の HTML ライブラリ 以下から入手できます。ftp://ftp.scriptics.com/pub/tcl/misc

```
#!/bin/sh
# the next line restarts using wish \
exec wish "$0" "$@"
# HTML ライブラリの読み込み
source [file dirname [info script]]/html_library.tcl

# 初期設定
proc Init {} {
    global w
    set w(size) 0
    set w(indent) 1.2
    # set w(home) [ file dirname [ info script ] ]/index.html
    set w(home) http://cosmo.ulis.ac.jp/~yuka/wkappa/index.html
    set w(url) $w(home)
    set w(message) "";
}
# 文書の読み込み
proc LoadFile {} {
    global w
    set ftypes {
        {"Text File"    { .htm .html } }
        {"All files"    *}
    }
    set fname [tk_getOpenFile -filetypes $ftypes -parent .]
    if {$fname == ""} {
        return
    } else {
        Disp_Html $fname
    }
}
```

```

}
# メニューバー
proc MakeMenu {} {
    global w
    # メニューバー用フレーム
    set w(menu) .menu
    frame $w(menu) -borderwidth 2 -relief raised
    pack $w(menu) -side top -fill x
    # メニュー
    set w(file) $w(menu).file
    menubutton $w(file) -text "ファイル" -menu $w(file).m
    pack $w(file) -side left
    # ファイル
    menu $w(file).m -tearoff no
    $w(file).m add command -label "ファイルを指定して開く" -command LoadFile
    $w(file).m add separator
    $w(file).m add command -label "終了" -command exit
}
# エントリ
proc MakeEntry {} {
    global w
    # エントリのフレーム
    set w(sub1) .fra1
    frame $w(sub1) -borderwidth 2 -relief ridge
    pack $w(sub1) -side top -fill x -expand no
    # エントリ
    set w(label) $w(sub1).lab
    label $w(label) -text "URL"
    set w(ent) $w(sub1).ent
    entry $w(ent) -textvariable w(url) -bg white
    pack $w(label) -side left
    pack $w(ent) -side left -fill x -expand yes
    bind $w(ent) <Key-Return> {
Hmlink_callback $w(text) $w(url)
    }
}
# テキスト
proc MakeText {} {
    global w
    # テキストのフレーム
    set w(main) .fra2
    frame $w(main) -borderwidth 2 -relief sunken
    pack $w(main) -side top -fill both -expand yes
    # テキストとスクロールバー
    set w(text) $w(main).txt
    text $w(text) -cursor hand2 -yscrollcommand "$w(main).scrl set"

```

```

    scrollbar $w(main).scrl -command "$w(text) yview"
    pack $w(main).scrl -side right -fill y
    pack $w(text) -side left -fill both -expand yes
}
# メッセージ
proc MakeMessage {} {
    global w
    # メッセージのフレーム
    set w(sub2) .fra3
    frame $w(sub2) -borderwidth 2 -relief ridge
    pack $w(sub2) -side top -fill both -expand no
    set w(mes) $w(sub2).mes
    label $w(mes) -textvariable w(message)
    pack $w(mes) -side left -fill both -expand yes
}
# HTML 文書の表示
proc Disp_Html { file } {
    global w HM.text
    $w(text) config -state normal
    set fragment ""
    regexp {[^#]*}#(.+)} $file dummy file fragment
    if { $file == "" && $fragment != "" } {
HMgoto $w(text) $fragment
    $w(text) config -state disabled
    return
    }
    HMreset_win $w(text)
    set w(message) "$file を表示しています。"
    update idletasks
    if { $fragment != "" } {
HMgoto $w(text) $fragment
    }
    set w(url) $file
    HMparse_html [ Load_Html $file ] "HMrender $w(text)"
    HMset_state $w(text) -stop 1
    set w(message) ""
    $w(text) config -state disabled
}
# HTML ファイルの読み込み
proc Load_Html { file } {
    global w

    if {[ string match http://* $file ]} {
# file is HTTP-accessible on network.
package require http
puts "Retrieving $file ..."

```

```

set token [::http::geturl $file]
::http::wait $token
puts "done"
return [::http::data $token]
}
if { [ catch { set fileid [ open $file ] } msg ] } {
return "
<title>$file は形式が間違っています。</title>
<h1>$file の読み込みエラーです。</h1><p>
$msg<hr>
<a href=$w(home)>Go home</a>
"
}
set result [ read $fileid ]
close $fileid
return $result
}
# リンク先へ移動
proc Hmlink_callback { win href } {
global w
if { [ string match http://* $href ] } {
# puts "HTTP Protocol Not support !!"
# exit
set w(url) $href
update
Disp_Html $w(url)
return
}
if { [string match z39.50s://* $href] } {
exec kappa $href &
return
}
if { [ string match #* $href ] } {
Disp_Html $href
return
}
if { [ string match /* $href ] } {
set w(url) $href
} else {
set w(url) [ file dirname $w(url) ]/$href
}
update
Disp_Html $w(url)
}
# イメージの処理
proc HMset_image {win handle src} {

```

```

    global w
    if { [ string match /* $src ] } {
set image $src
    } else {
set image [ file dirname $w(url) ]/$src
    }
    set w(message) "イメージ $image を読み込んでいます。"
    update
    if { [string first " $image " " [image names] " ] >= 0 } {
HMgot_image $handle $image
    } else {
set type photo
if { [ file extension $image ] == ".bmp" } { set type bitmap }
catch { image create $type $image -file $image } image
HMgot_image $handle $image
    }
}
# -----
#   メイン
# -----
Init
MakeMenu
MakeEntry
MakeText
MakeMessage
#
HMinet_win $w(text)
HMset_state $w(text) -size $w(size)
HMset_indent $w(text) $w(indent)
Disp_Html $w(home)
#

```


付録 D

Z39.50 クライアントのプログラム

Kappa のインストールには Tcl/Tk ライブラリ、YAZ ライブラリが必要である。

Tcl/Tk ライブラリ (Tcl/Tk 8.05jp) 日本語化されているものをインストールしないと日本語の入力表示ができません。以下から入手できる。 <ftp://ftp.sra.co.jp/pub/lang/tcl/>

YAZ ライブラリ (YAZ-1.4pl2) インストールする必要はありませんが、コンパイルは必要である。以下から入手できる。 <http://www.indexdata.dk/yaz/>

D.1 client.c

```
/*
 * 1999.3.27
 * ヒットした件数が 30 件以上だったら show エントリには 30 を入力するように変更
 *
 * 1997.10.14
 * Usmarc は、リストボックスに表示できるように
 *
 */

/*
 * This is the obligatory little toy client, whose primary purpose is
 * to illustrate the use of the YAZ service-level API.
 */

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <assert.h>
#include <unistd.h>
#include <stdarg.h>

/* YAZ Library */
#include <yaz-util.h>

#include <comstack.h>
#include <tcpip.h>
#ifdef USE_XTIMOSI
#include <xmosi.h>
#endif

#include <proto.h>
#include <marcdisp.h>
#include <diagbib1.h>

#include <pquery.h>

#if CCL2RPN
#include <yaz-ccl.h>
#endif

/* Tcl/Tk */
```

```

#include <tcl.h>
#include <tk.h>

#include "config.h"

#define TMPDIR "/tmp"
#define BUFSIZE 1024

/* Tcl/Tk Callback Function */
static char *TclCmdName = "TclClientCommand";
int tclCommandCallback(ClientData cd, int argc, char *argv[]);

void tclPrintf(char *str, int lnum);
Tcl_Interp *interp;
FILE *tcldump;
char tcldumpfilename[BUFSIZE];
static int show_start_position;
int buf_setnumber; /* show の時に必要 */

static ODR out, in, print; /* encoding and decoding streams */
static COMSTACK conn = 0; /* our z-association */
static Z_IdAuthentication *auth = 0; /* our current auth definition */
static char *databaseNames[BUFSIZE];
static int num_databaseNames = 0;
static int setnumber = 0; /* current result set number */
static int smallSetUpperBound = 0;
static int largeSetLowerBound = 1;
static int mediumSetPresentNumber = 0;
static Z_ElementSetNames *elementSetNames = 0;
static int setno = 1; /* current set offset */
static int protocol = PROTO_Z3950; /* current app protocol */
static int recordsyntax = VAL_USMARC;
static int sent_close = 0;
static ODR_MEM session_mem; /* memory handle for init-response */
static Z_InitResponse *session = 0; /* session parameters */
static char last_scan[BUFSIZE] = "0";
static FILE *marcdump = 0;

typedef enum {
    QueryType_Prefix,
    QueryType_CCL,
    QueryType_CCL2RPN
} QueryType;

/* static QueryType queryType = QueryType_Prefix; */
static QueryType queryType = QueryType_CCL2RPN;

```

```

#if CCL2RPN
static CCL_bibset bibset;          /* CCL bibset handle */
#endif

static void send_apdu(Z_APDU *a)
{
    char *buf;
    int len;

    if (!z_APDU(out, &a, 0)) {
        odr_perror(out, "Encoding APDU");
        exit(1);
    }
    buf = odr_getbuf(out, &len, 0);
    odr_reset(out); /* release the APDU structure */
    if (cs_put(conn, buf, len) < 0) {
        fprintf(stderr, "cs_put: %s", cs_errmsg(cs_errno(conn)));
        exit(1);
    }
}

/* INIT SERVICE ----- */

static void send_initRequest()
{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_initRequest);
    Z_InitRequest *req = apdu->u.initRequest;

    ODR_MASK_SET(req->options, Z_Options_search);
    ODR_MASK_SET(req->options, Z_Options_present);
    ODR_MASK_SET(req->options, Z_Options_namedResultSets);
    ODR_MASK_SET(req->options, Z_Options_triggerResourceCtrl);
    ODR_MASK_SET(req->options, Z_Options_scan);
    ODR_MASK_SET(req->options, Z_Options_sort);

    ODR_MASK_SET(req->protocolVersion, Z_ProtocolVersion_1);
    ODR_MASK_SET(req->protocolVersion, Z_ProtocolVersion_2);
    ODR_MASK_SET(req->protocolVersion, Z_ProtocolVersion_3);

    *req->maximumRecordSize = 1024*1024;

    req->implementationName = PACKAGE;
    req->implementationVersion = VERSION;

    req->idAuthentication = auth;
}

```

```

    send_apdu(apdu);
    printf("Sent initrequest.\n");
}

static int process_initResponse(Z_InitResponse *res)
{
    /* save session parameters for later use */
    session_mem = odr_extract_mem(in);
    session = res;

    if (!*res->result)
        printf("Connection rejected by target.\n");
    else
        printf("Connection accepted by target.\n");
    if (res->implementationId)
        printf("ID      : %s\n", res->implementationId);
    if (res->implementationName)
        printf("Name   : %s\n", res->implementationName);
    if (res->implementationVersion)
        printf("Version: %s\n", res->implementationVersion);
    if (res->userInformationField) {
        printf("UserInformationfield:\n");
        if (!z_External(print, (Z_External*)&res-> userInformationField, 0)) {
            odr_perror(print, "Printing userinfo\n");
            odr_reset(print);
        }
        if (res->userInformationField->which == Z_External_octet) {
            printf("Guessing visiblestring:\n");
            printf("'%'s'\n", res->userInformationField->u. octet_aligned->buf);
        }
    }
}
return 0;
}

int cmd_open(char *arg)
{
    void *add;
    char type[BUFSIZE], addr[BUFSIZE];
    CS_TYPE t;
    char tcl_buf[BUFSIZE];

    if (conn) {
        /* printf("Already connected.\n");*/
        tclPrintf("Already connected.", 4);
        return 0;
    }
}

```

```

}
if (!*arg || sscanf(arg, "%[^:]:%s", type, addr) < 2) {
    /*fprintf(stderr, "Usage: open (osi|tcp) ':' [tsel '/' ]host[':'port]\n");*/
tclPrintf("Usage: open (osi|tcp) ':' [tsel '/' ]host[':'port]", 4);
    return 0;
}
if (!strcmp(type, "tcp")) {
t = tcpip_type;
protocol = PROTO_Z3950;
}
else
#ifdef USE_XTIMOSI
if (!strcmp(type, "osi")) {
    t = mosi_type;
    protocol = PROTO_SR;
}
else
#endif
{
    sprintf(tcl_buf, "Bad type: %s\n", type);
    tclPrintf(tcl_buf, 4);
    return 0;
}
if (!(conn = cs_create(t, 1, protocol))) {
    perror("cs_create");
tclPrintf("Cannot connect", 4);
    return 0;
}
if (!(add = cs_straddr(conn, addr))) {
perror(arg);
tclPrintf("cannot open", 4);
return 0;
}
printf("Connecting...");
fflush(stdout);
if (cs_connect(conn, add) < 0) {
    perror("connect");
tclPrintf("connection error occured", 4);
    cs_close(conn);
    conn = 0;
    return 0;
}
printf("Ok.\n");
send_initRequest();
return 2;
}

```

```

int cmd_authentication(char *arg)
{
    static Z_IdAuthentication au;
    static char open[BUFSIZE];

    if (!*arg) {
        printf("Auth field set to null\n");
        auth = 0;
        return 1;
    }
    auth = &au;
    au.which = Z_IdAuthentication_open;
    au.u.open = open;
    strcpy(open, arg);
    return 1;
}

/* SEARCH SERVICE ----- */

static void display_variant(Z_Variant *v, int level, FILE *fp)
{
    int i;

    for (i = 0; i < v->num_triples; i++) {
        fprintf(fp, "%*sclass=%d,type=%d", level * 4, "",
*v->triples[i]->zclass,
*v->triples[i]->type);
if (v->triples[i]->which == Z_Triple_internationalString)
    fprintf(fp, ",value=%s\n", v->triples[i]->value.internationalString);
else
    fprintf(fp, "\n");
    }
}

static void display_grs1(Z_GenericRecord *r, int level, FILE *fp)
{
    int i;

    if (!r)
        return;
    for (i = 0; i < r->num_elements; i++) {
        Z_TaggedElement *t;

        fprintf(fp, "%*s", level * 4, "");
        t = r->elements[i];
    }
}

```

```

fprintf(fp, "(");
if (t->tagType)
    fprintf(fp, "%d,", *t->tagType);
else
    fprintf(fp, "?");
if (t->tagValue->which == Z_StringOrNumeric_numeric)
    fprintf(fp, "%d ", *t->tagValue->u.numeric);
else
    fprintf(fp, "%s ", t->tagValue->u.string);
if (t->content->which == Z_ElementData_subtree) {
    fprintf(fp, "\n");
    display_grs1(t->content->u.subtree, level+1, fp);
}
else if (t->content->which == Z_ElementData_string)
    fprintf(fp, "%s\n", t->content->u.string);
else if (t->content->which == Z_ElementData_numeric)
    fprintf(fp, "%d\n", *t->content->u.numeric);
else if (t->content->which == Z_ElementData_oid) {
    int *ip = t->content->u.oid;
    oident *oent;

    if ((oent = oid_getentbyoid(t->content->u.oid)))
        fprintf(fp, "OID: %s\n", oent->desc);
    else {
        fprintf(fp, "{");
        while (ip && *ip >= 0)
            fprintf(fp, " %d", *(ip++));
        fprintf(fp, " }\n");
    }
}
else if (t->content->which == Z_ElementData_noDataRequested)
    fprintf(fp, "[No data requested]\n");
else if (t->content->which == Z_ElementData_elementEmpty)
    fprintf(fp, "[Element empty]\n");
else if (t->content->which == Z_ElementData_elementNotThere)
    fprintf(fp, "[Element not there]\n");
else
    fprintf(fp, "??????\n");
if (t->appliedVariant)
    display_variant(t->appliedVariant, level+1, fp);
if (t->metaData && t->metaData->supportedVariants) {
    int c;

    fprintf(fp, "%s---- variant list\n", (level+1)*4, "");
    for (c = 0; c < t->metaData->num_supportedVariants; c++) {
        fprintf(fp, "%svariant #%d\n", (level+1)*4, "", c);
    }
}

```



```

display_variant(t->metaData->supportedVariants[c], level + 2,fp);
    }
}
}
}

static void display_record(Z_DatabaseRecord *p)
{
    Z_External *r = (Z_External*) p;
    oident *ent = oid_getentbyoid(r->direct_reference);

    /*
     * Tell the user what we got.
     */
    if (r->direct_reference) {
        /* printf("Record type: ");*/
        fprintf(tcldump,"Record type:");
    if (ent)
        /*printf("%s\n", ent->desc);*/
        fprintf(tcldump,"%s\n--\n", ent->desc);
        else if (!odr_oid(print, &r->direct_reference, 0)) {
            odr_perror(print, "print oid");
            odr_reset(print);
        }
    }
    /* Check if this is a known, ASN.1 type tucked away in an octet string */
    if (ent && r->which == Z_External_octet) {
        Z_ext_typeent *type = z_ext_getentbyref(ent->value);
        void *rr;

    if (type) {
        /*
         * Call the given decoder to process the record.
         */
        odr_setbuf(in, (char*)p->u.octet_aligned->buf,
            p->u.octet_aligned->len, 0);
        if (!(*type->fun)(in, &rr, 0)) {
            odr_perror(in, "Decoding constructed record.");
            fprintf(stderr, "[Near %d]\n", odr_offset(in));
            fprintf(stderr, "Packet dump:\n-----\n");
            odr_dumpBER(stderr, (char*)p->u.octet_aligned->buf,
                p->u.octet_aligned->len);
            fprintf(stderr, "-----\n");
            exit(1);
        }
        /*

```

```

    * Note: we throw away the original, BER-encoded record here.
    * Do something else with it if you want to keep it.
    */
r->u.sutrs = rr;    /* we don't actually check the type here. */
r->which = type->what;
}
}
if (ent && ent->value == VAL_SOIF)
    printf("%.s", r->u.octet_aligned->len, r->u.octet_aligned->buf);
else if (r->which == Z_External_octet && p->u.octet_aligned->len) {
    const char *marc_buf = (char*)p->u.octet_aligned->buf;
    marc_display (marc_buf, tcldump);
    if (marcdump)
        fwrite(marc_buf, strlen (marc_buf), 1, marcdump);
}
else if (ent && ent->value == VAL_SUTRS) {
    if (r->which != Z_External_sutrs) {
        printf("Expecting single SUTRS type for SUTRS.\n");
        return;
    }
    fprintf(tcldump, "%.s\n", r->u.sutrs->len, r->u.sutrs->buf);
}
else if (ent && ent->value == VAL_GRS1) {
    if (r->which != Z_External_grs1) {
        printf("Expecting single GRS type for GRS.\n");
        return;
    }
    display_grs1(r->u.grs1, 0, tcldump);
}
else {
    printf("Unknown record representation.\n");
    if (!z_External(print, &r, 0)) {
        odr_perror(print, "Printing external");
        odr_reset(print);
    }
}
}

static void display_diagreCs(Z_DiagRec **pp, int num)
{
    int i;
    oident *ent;
    Z_DefaultDiagFormat *r;
    char buf[BUFSIZE], buf2[BUFSIZE];

    strcpy(buf, "Diagnostic message(s) from database:\n");

```

```

    for (i = 0; i<num; i++) {
Z_DiagRec *p = pp[i];
if (p->which != Z_DiagRec_defaultFormat) {
    strcat(buf, "Diagnostic record not in default format.\n");
    tclPrintf(buf, 2);
    return;
} else {
    r = p->u.defaultFormat;
}
if (!(ent = oid_getentbyoid(r->diagnosticSetId)) ||
    ent->oclass != CLASS_DIAGSET || ent->value != VAL_BIB1)
    printf("Missing or unknown diagset\n");
sprintf(buf2, " [%d] %s", *r->condition, diagbib1_str(*r->condition));
strcat(buf, buf2);
if (r->addinfo && *r->addinfo) {
    sprintf(buf2, " -- '%s'\n", r->addinfo);
    strcat(buf, buf2);
} else {
    strcat(buf, "\n");
}
    tclPrintf(buf, 2);
}

static void display_nameplusrecord(Z_NamePlusRecord *p)
{
    if (p->databaseName)
fprintf(tcldump, "DBname:%s  ", p->databaseName);
    if (p->which == Z_NamePlusRecord_surrogateDiagnostic)
        display_diagreCs(&p->u.surrogateDiagnostic, 1);
    else
        display_record(p->u.databaseRecord);
}

static void display_records(Z_Records *p)
{
    int i;
    char buf[BUFSIZE];
    char nkf_command[BUFSIZE];

    if (p->which == Z_Records_NSD)
display_diagreCs (&p->u.nonSurrogateDiagnostic, 1);
        else if (p->which == Z_Records_multipleNSD)
display_diagreCs (p->u.multipleNonSurDiagnostics->diagRecs,
    p->u.multipleNonSurDiagnostics->num_diagRecs);
        else {

```

```

tcldump = fopen(tcldumpfilename, "w");

fprintf(tcldump, "***** \n   %d 件のレ
コードが返ってきました\n", p->u.databaseOrSurDiagnostics->num_records);

    for (i = 0; i < p->u.databaseOrSurDiagnostics->num_records; i++) {
        fprintf(tcldump, "-----
\n[%d] No%d   ",
            buf_setnumber, show_start_position++);
            display_nameplusrecord(p->u.databaseOrSurDiagnostics->records[i]);
    }
    }
fclose(tcldump);
sprintf(nkf_command, "nkf -e %s > %s_EUC" ,tcldumpfilename ,tcldumpfilename);
system(nkf_command);
sprintf(buf, "recordToListbox %s_EUC .l1", tcldumpfilename);
Tcl_Eval(interp, buf);
}

static int send_searchRequest(char *arg)
{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_searchRequest);
    Z_SearchRequest *req = apdu->u.searchRequest;
    Z_Query query;
    int oid[OID_SIZE];
#if CCL2RPN
    struct ccl_rpn_node *rpn;
    int error, pos;
    oident bib1;
#endif
    char setstring[100];
    Z_RPNQuery *RPNquery;
    Odr_oct ccl_query;
    char err_buf[256];

#if CCL2RPN
    if (queryType == QueryType_CCL2RPN) {
        rpn = ccl_find_str(bibset, arg, &error, &pos);
        if (error) {
            sprintf(err_buf, "CCL ERROR: %s\n", ccl_err_msg(error));
            tclPrintf(err_buf, 2);
            return 0;
        }
    }
}
#endif

```

```

if (!strcmp(arg, "@big")) { /* strictly for troublemaking */
    static unsigned char big[2100];
    static Odr_oct bigo;

    /* send a very big referenceid to test transport stack etc. */
    memset(big, 'A', 2100);
    bigo.len = bigo.size = 2100;
    bigo.buf = big;
    req->referenceId = &bigo;
}

if (setnumber >= 0) {
    sprintf(setstring, "%d", ++setnumber);
    req->resultSetName = setstring;
}
*req->smallSetUpperBound = smallSetUpperBound;
*req->largeSetLowerBound = largeSetLowerBound;
*req->mediumSetPresentNumber = mediumSetPresentNumber;
if (smallSetUpperBound > 0 ||
(largeSetLowerBound > 1 && mediumSetPresentNumber > 0)) {

oident prefsyn;

    prefsyn.proto = protocol;
    prefsyn.oclass = CLASS_RECSYN;
    prefsyn.value = recordsyntax;
    req->preferredRecordSyntax =
odr_oiddup(out, oid_ent_to_oid(&prefsyn, oid));
    req->smallSetElementSetNames = req->mediumSetElementSetNames = elementSetNames;
}
req->num_databaseNames = num_databaseNames;
req->databaseNames = databaseNames;

req->query = &query;

switch (queryType) {
case QueryType_Prefix:
    query.which = Z_Query_type_1;
    RPNquery = p_query_rpn (out, protocol, arg);
    if (!RPNquery) {
        tclPrintf("Prefix query error\n", 2);
        return 0;
    }
    query.u.type_1 = RPNquery;
    break;
case QueryType_CCL:

```

```

        query.which = Z_Query_type_2;
        query.u.type_2 = &ccl_query;
        ccl_query.buf = (unsigned char*) arg;
        ccl_query.len = strlen(arg);
        break;
#if CCL2RPN
    case QueryType_CCL2RPN:
        query.which = Z_Query_type_1;
        assert((RPNquery = ccl_rpn_query(out, rpn)));
        bib1.proto = protocol;
        bib1.oclass = CLASS_ATTSET;
        bib1.value = VAL_BIB1;
        RPNquery->attributeSetId = oid_ent_to_oid(&bib1, oid);
        query.u.type_1 = RPNquery;
        break;
#endif
    default:
        tclPrintf ("Unsupported query type\n", 2);
        return 0;
}
send_apdu(apdu);
setno = 1;
printf("Sent searchRequest.\n");
return 2;
}

static int process_searchResponse(Z_SearchResponse *res)
{
    char buf[BUFSIZE];

    if (*res->searchStatus)
        printf("Search was a success.\n");
    else
        printf("Search was failed.\n");
    /* 10.12 process_searchResponse() */
    /* リストボックスに、検索式とヒットした件数を表示し */
    /* show のためのエントリウィジットにその値を挿入 */
    sprintf(buf, ".l0 insert end {[%3d] 検索式: %s   ヒットした件数:%d}\n",
        setnumber, Tcl_GetVar(interp, "TclEntryArg", 0), *res->resultCount);
    Tcl_Eval(interp, buf);

    strcpy(buf, ".showEnter.setno delete 0 end\n");
    strcat(buf, ".showEnter.beginno delete 0 end\n");
    strcat(buf, ".showEnter.endno delete 0 end\n");
    strcat(buf, ".showEnter.setno delete 0 end\n");
    strcat(buf, ".showEnter.beginno delete 0 end\n");
}

```

```

strcat(buf, ".showEnter.endno delete 0 end" );
Tcl_Eval(interp, buf);

sprintf(buf, ".showEnter.setno insert 0 {%d}", setnumber);
Tcl_Eval(interp, buf);
strcpy(buf, ".showEnter.beginno insert 0 1");
Tcl_Eval(interp, buf);

/* ヒットした件数が 30 件以上だったら show エントリには 30 を入力 */
sprintf(buf, ".showEnter.endno insert 0 {%d}",
*res->resultCount > 30 ? 30 : *res->resultCount);
Tcl_Eval(interp, buf);

printf("records returned: %d\n", *res->numberOfRecordsReturned);
setno += *res->numberOfRecordsReturned;
if (res->records)
    display_records(res->records);
return 0;
}

int cmd_find(char *arg)
{
    if (!*arg) {
        tclPrintf("Find what?\n", 2);
        return 0;
    }
    if (!conn) {
        printf("Not connected yet\n");
        return 0;
    }
    if (!send_searchRequest(arg))
        return 0;
    return 2;
}

int cmd_ssub(char *arg)
{
    if (!(smallSetUpperBound = atoi(arg)))
        return 0;
    return 1;
}

int cmd_lslb(char *arg)
{
    if (!(largeSetLowerBound = atoi(arg)))
        return 0;
}

```

```

    return 1;
}

int cmd_mspn(char *arg)
{
    if (!(mediumSetPresentNumber = atoi(arg)))
        return 0;
    return 1;
}

int cmd_status(char *arg)
{
    printf("smallSetUpperBound: %d\n", smallSetUpperBound);
    printf("largeSetLowerBound: %d\n", largeSetLowerBound);
    printf("mediumSetPresentNumber: %d\n", mediumSetPresentNumber);
    return 1;
}

int cmd_base(char *arg)
{
    int i;
    char *cp;

    if (!*arg) {
        printf("Usage: base <database> <database> ... \n");
        return 0;
    }
    for (i = 0; i < num_databaseNames; i++)
        xfree (databaseNames[i]);
    num_databaseNames = 0;
    while (1) {
        if (!(cp = strchr(arg, ' ')))
            cp = arg + strlen(arg);
        if (cp - arg < 1)
            break;
        databaseNames[num_databaseNames] = xmalloc (1 + cp - arg);
        memcpy (databaseNames[num_databaseNames], arg, cp - arg);
        databaseNames[num_databaseNames++][cp - arg] = '\0';
        if (!*cp)
            break;
        arg = cp+1;
    }
    return 1;
}

int cmd_setnames(char *arg)

```



```

{
    if (setnumber < 0) {
        printf("Set numbering enabled.\n");
        setnumber = 0;
    }
    else {
        printf("Set numbering disabled.\n");
        setnumber = -1;
    }
    return 1;
}

/* PRESENT SERVICE ----- */

static int send_presentRequest(char *arg)
{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_presentRequest);
    Z_PresentRequest *req = apdu->u.presentRequest;
    Z_RecordComposition compo;
    oident prefsyn;
    int nos = 1;
    int oid[OID_SIZE];
    char *p;
    char setstring[100];

    /* *** input 1997.9.9 send_presentRequest() */
    char *p2;

    if((p2 = strchr(arg, '@')) {
buf_setnumber = atoi(p2 + 1);
printf("%d\n",buf_setnumber);
*p2 = 0;
    } else {
buf_setnumber = setnumber;
    }
    /* *** end input 1997.9.9 */

    if ((p = strchr(arg, '+')) {
        nos = atoi(p + 1);
        *p = 0;
    }

    if (*arg) {
        setno = atoi(arg);
show_start_position = setno;
    }
}

```

```

    if (p && (p=strchr(p+1, '+'))) {
        strcpy (setstring, p+1);
        req->resultSetId = setstring;
    } else if (setnumber >= 0) {
/* *** input 1997.9.9 send_presentRequest() */

sprintf(setstring, "%d", buf_setnumber);
/* *** end input 1997.9.9 */

        req->resultSetId = setstring;
    }
#if 0
    if (1)
static Z_Range range;
    static Z_Range *rangep = &range;
    req->num_ranges = 1;
#endif
    req->resultSetStartPoint = &setno;
    req->numberOfRecordsRequested = &nos;
    prefsyn.proto = protocol;
    prefsyn.oclass = CLASS_RECSYN;
    prefsyn.value = recordsyntax;
    req->preferredRecordSyntax = oid_ent_to_oid(&prefsyn, oid);

    if (elementSetNames) {
        req->recordComposition = &compo;
        compo.which = Z_RecordComp_simple;
        compo.u.simple = elementSetNames;
    }

    send_apdu(apdu);

    printf("Sent presentRequest (%d+%d).\n", setno, nos);
    return 2;
}

void process_close(Z_Close *req)
{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_close);
    Z_Close *res = apdu->u.close;

    static char *reasons[] = {
        "finished",
        "shutdown",
        "system problem",
    }

```

```

        "cost limit reached",
        "resources",
        "security violation",
        "protocolError",
        "lack of activity",
        "peer abort",
        "unspecified"
    };

    printf("Reason: %s, message: %s\n", reasons[*req->closeReason],
    req->diagnosticInformation ? req->diagnosticInformation : "NULL");
    if (sent_close) {
printf("Goodbye.\n");
unlink(tcldumpfilename);
exit(1);
    }

    *res->closeReason = Z_Close_finished;
    send_apdu(apdu);
    printf("Sent response.\n");
    sent_close = 1;
}

int cmd_show(char *arg)
{
    if (!session) {
        printf("Session not initialized yet\n");
        return 0;
    }
    if (!send_presentRequest(arg)) {
        return 0;
    }
    return 2;
}

int cmd_quit(char *arg)
{
    printf("See you later, alligator.\n");
    unlink(tcldumpfilename);
    strcat(tcldumpfilename, "_EUC");
    unlink(tcldumpfilename);
    exit(0);
    return 0;
}

int cmd_cancel(char *arg)

```

```

{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_triggerResourceControlRequest);
    Z_TriggerResourceControlRequest *req =
        apdu->u.triggerResourceControlRequest;
    bool_t rfalse = 0;

    if (!session) {
        printf("Session not initialized yet\n");
        return 0;
    }
    if (!ODR_MASK_GET(session->options, Z_Options_triggerResourceCtrl)) {
        printf("Target doesn't support cancel (trigger resource ctrl)\n");
        return 0;
    }
    *req->requestedAction = Z_TriggerResourceCtrl_cancel;
    req->resultSetWanted = &rfalse;

    send_apdu(apdu);
    printf("Sent cancel request\n");
    return 2;
}

int send_scanrequest(char *string, int pp, int num)
{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_scanRequest);
    Z_ScanRequest *req = apdu->u.scanRequest;

    req->num_databaseNames = num_databaseNames;
    req->databaseNames = databaseNames;
    req->termListAndStartPoint = p_query_scan(out, protocol,
        &req->attributeSet, string);
    req->numberOfTermsRequested = &num;
    req->preferredPositionInResponse = &pp;
    send_apdu(apdu);
    return 2;
}

int send_sortrequest(char *arg, int newset)
{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_sortRequest);
    Z_SortRequest *req = apdu->u.sortRequest;
    Z_SortKeySpecList *sksl = odr_malloc (out, sizeof(*sksl));
    char setstring[32];
    char sort_string[32], sort_flags[32];
    int off;
    int oid[OID_SIZE];

```

```

    oident bib1;

    if (setnumber >= 0)
sprintf (setstring, "%d", setnumber);
    else
sprintf (setstring, "default");

    req->inputResultSetNames =
odr_malloc (out, sizeof(*req->inputResultSetNames));
    req->inputResultSetNames->num_strings = 1;
    req->inputResultSetNames->strings =
odr_malloc (out, sizeof(*req->inputResultSetNames->strings));
    req->inputResultSetNames->strings[0] =
odr_malloc (out, strlen(setstring)+1);
    strcpy (req->inputResultSetNames->strings[0], setstring);

    if (newset && setnumber >= 0)
sprintf (setstring, "%d", ++setnumber);

    req->sortedResultSetName = odr_malloc (out, strlen(setstring)+1);
    strcpy (req->sortedResultSetName, setstring);

    req->sortSequence = sksl;
    sksl->num_specs = 0;
    sksl->specs = odr_malloc (out, sizeof(sksl->specs) * 20);

    bib1.proto = protocol;
    bib1.oclass = CLASS_ATTSET;
    bib1.value = VAL_BIB1;
    while ((sscanf (arg, "%31s %31s%n", sort_string, sort_flags, &off)) == 2
&& off > 1) {
int i;
char *sort_string_sep;
Z_SortKeySpec *sks = odr_malloc (out, sizeof(*sks));
Z_SortKey *sk = odr_malloc (out, sizeof(*sk));

arg += off;
sksl->specs[sksl->num_specs++] = sks;
sks->sortElement = odr_malloc (out, sizeof(*sks->sortElement));
sks->sortElement->which = Z_SortElement_generic;
sks->sortElement->u.generic = sk;

if ((sort_string_sep = strchr (sort_string, '=')) {
    Z_AttributeElement *el = odr_malloc (out, sizeof(*el));
    sk->which = Z_SortKey_sortAttributes;
    sk->u.sortAttributes =

```

```

odr_malloc (out, sizeof(*sk->u.sortAttributes));
    sk->u.sortAttributes->id = oid_ent_to_oid(&bib1, oid);
    sk->u.sortAttributes->list =
odr_malloc (out, sizeof(*sk->u.sortAttributes->list));
    sk->u.sortAttributes->list->num_attributes = 1;
    sk->u.sortAttributes->list->attributes =
odr_malloc (out,
    sizeof(*sk->u.sortAttributes->list->attributes));
    sk->u.sortAttributes->list->attributes[0] = el;
    el->attributeSet = 0;
    el->attributeType = odr_malloc (out, sizeof(*el->attributeType));
    *el->attributeType = atoi (sort_string);
    el->which = Z_AttributeValue_numeric;
    el->value.numeric = odr_malloc (out, sizeof(*el->value.numeric));
    *el->value.numeric = atoi (sort_string_sep + 1);
} else {
    sk->which = Z_SortKey_sortField;
    sk->u.sortField = odr_malloc (out, strlen(sort_string)+1);
    strcpy (sk->u.sortField, sort_string);
}
sks->sortRelation = odr_malloc (out, sizeof(*sks->sortRelation));
*sks->sortRelation = Z_SortRelation_ascending;
sks->caseSensitivity = odr_malloc (out, sizeof(*sks->caseSensitivity));
*sks->caseSensitivity = Z_SortCase_caseSensitive;

sks->missingValueAction = NULL;

for (i = 0; sort_flags[i]; i++) {
    switch (sort_flags[i]) {
        case 'a':
        case 'A':
        case '>':
*sks->sortRelation = Z_SortRelation_ascending;
break;
        case 'd':
        case 'D':
        case '<':
*sks->sortRelation = Z_SortRelation_descending;
break;
        case 'i':
        case 'I':
*sks->caseSensitivity = Z_SortCase_caseInsensitive;
break;
        case 'S':
        case 's':
*sks->caseSensitivity = Z_SortCase_caseSensitive;

```

```

break;
    }
}
    }
    if (!skssl->num_specs) {
        printf ("Missing sort specifications\n");
return -1;
    }
    send_apdu(apdu);
    return 2;
}

void display_term(Z_TermInfo *t)
{
    if (t->term->which == Z_Term_general) {
        printf("%.*s (%d)\n", t->term->u.general->len, t->term->u.general->buf,
            t->globalOccurrences ? *t->globalOccurrences : -1);
        sprintf(last_scan, "%.*s", t->term->u.general->len,
t->term->u.general->buf);
    } else {
        printf("Term type not general.\n");
    }
}

void process_scanResponse(Z_ScanResponse *res)
{
    int i;

    printf("SCAN: %d entries, position=%d\n", *res->numberOfEntriesReturned,
*res->positionOfTerm);
    if (*res->scanStatus != Z_Scan_success)
        printf("Scan returned code %d\n", *res->scanStatus);
    if (!res->entries)
        return;
    if (res->entries->which == Z_ListEntries_entries) {
        Z_Entries *ent = res->entries->u.entries;

        for (i = 0; i < ent->num_entries; i++)
            if (ent->entries[i]->which == Z_Entry_termInfo) {
                printf("%c ", i + 1 == *res->positionOfTerm ? '*' : ' ');
                display_term(ent->entries[i]->u.termInfo);
            } else
                display_diagreCs(&ent->entries[i]->u.surrogateDiagnostic, 1);
    }
    else
        display_diagreCs(&res->entries->

```

```

    u.nonSurrogateDiagnostics->diagRecs[0], 1);
}

void process_sortResponse(Z_SortResponse *res)
{
    printf("Sort: status=");
    switch (*res->sortStatus) {
        case Z_SortStatus_success:
printf ("success"); break;
        case Z_SortStatus_partial_1:
printf ("partial"); break;
        case Z_SortStatus_failure:
printf ("failure"); break;
        default:
printf ("unknown (%d)", *res->sortStatus);
    }
    printf ("\n");
    if (res->diagnostics)
        display_diagreCs(res->diagnostics->diagRecs,
res->diagnostics->num_diagRecs);
}

int cmd_sort_generic(char *arg, int newset)
{
    if (!session) {
        printf("Session not initialized yet\n");
        return 0;
    }
    if (!ODR_MASK_GET(session->options, Z_Options_sort)) {
        printf("Target doesn't support sort\n");
        return 0;
    }
    if (*arg) {
        if (send_sortrequest(arg, newset) < 0)
            return 0;
    }
return 2;
}

int cmd_sort(char *arg)
{
    return cmd_sort_generic (arg, 0);
}

int cmd_sort_newset (char *arg)

```



```

{
    return cmd_sort_generic (arg, 1);
}

int cmd_scan(char *arg)
{
    if (!session) {
        printf("Session not initialized yet\n");
        return 0;
    }
    if (!ODR_MASK_GET(session->options, Z_Options_scan)) {
        printf("Target doesn't support scan\n");
        return 0;
    }
    if (*arg) {
        if (send_scanrequest(arg, 5, 20) < 0)
            return 0;
        } else if (send_scanrequest(last_scan, 1, 20) < 0) {
return 0;
        }
        return 2;
    }

int cmd_format(char *arg)
{
    if (!arg || !*arg) {
        printf("Usage: format <recordsyntax>\n");
        return 0;
    }
    recordsyntax = oid_getvalbyname (arg);
    if (recordsyntax == VAL_NONE) {
        printf ("unknown record syntax\n");
        return 0;
    }
    return 1;
}

int cmd_elements(char *arg)
{
    static Z_ElementSetNames esn;
    static char what[100];

    if (!arg || !*arg) {
        printf("Usage: elements <esn>\n");
        return 0;
    }
}

```

```

    strcpy(what, arg);
    esn.which = Z_ElementSetNames_generic;
    esn.u.generic = what;
    elementSetNames = &esn;
    return 1;
}

int cmd_attributeset(char *arg)
{
    char what[100];

    if (!arg || !*arg) {
printf("Usage: attributeset <setname>\n");
return 0;
    }
    sscanf(arg, "%s", what);
    if (p_query_attset (what)) {
printf("Unknown attribute set name\n");
return 0;
    }
    return 1;
}

int cmd_querytype (char *arg)
{
    if (!strcmp (arg, "ccl"))
        queryType = QueryType_CCL;
    else if (!strcmp (arg, "prefix"))
        queryType = QueryType_Prefix;
#ifdef CCL2RPN
    else if (!strcmp (arg, "ccl2rpn") || !strcmp (arg, "cclrpn"))
        queryType = QueryType_CCL2RPN;
#endif
    else {
        printf ("Querytype must be one of:\n");
        printf (" prefix          - Prefix query\n");
        printf (" ccl              - CCL query\n");
#ifdef CCL2RPN
        printf (" ccl2rpn         - CCL query converted to RPN\n");
#endif
    }
    return 0;
}
return 1;
}

int cmd_close(char *arg)

```

```

{
    Z_APDU *apdu = zget_APDU(out, Z_APDU_close);
    Z_Close *req = apdu->u.close;

    *req->closeReason = Z_Close_finished;
    send_apdu(apdu);
    printf("Sent close request.\n");
    sent_close = 1;
    return 2;
}

static void initialize(void)
{
#ifdef CCL2RPN
    FILE *inf;
    char filename[BUFSIZE];
#endif
    nmem_init();
    if (!(out = odr_createmem(ODR_ENCODE)) ||
        !(in = odr_createmem(ODR_DECODE)) ||
        !(print = odr_createmem(ODR_PRINT))) {
        fprintf(stderr, "failed to allocate ODR streams\n");
        exit(1);
    }
    setvbuf(stdout, 0, _IONBF, 0);

#ifdef CCL2RPN
    bibset = ccl_qual_mk ();
    strcpy(filename, DATA_DIR);
    strcat(filename, "/default.bib");
    inf = fopen (filename, "r");
    if (inf) {
        ccl_qual_file (bibset, inf);
        fclose (inf);
    }
#endif
}

static int client(int argc, char **argv)
{
    Tk_Window toplevel;
    char inTclFilename[BUFSIZE];
    char buf[BUFSIZE];
    char addr[BUFSIZE], dbname[BUFSIZE], host[BUFSIZE], port[BUFSIZE];
    strcpy(inTclFilename, DATA_DIR);
    strcat(inTclFilename, "/client.tcl");

```

```

/* extracts from Tk_Main below */

/* Initialize Tcl interpreter */
Tcl_FindExecutable(argv[0]);
interp = Tcl_CreateInterp();

/* parse options here if needed */

if (Tcl_Init(interp) != TCL_OK) {
fprintf(stderr, "Tcl_Init failed: %s\n", interp->result);
}

/* Initialize Tk */
/* application name/class determined from "argv0" (not set here)*/
if (Tk_Init(interp) != TCL_OK) {
fprintf(stderr, "Tk_Init failed: %s\n", interp->result);
}
toplevel = Tk_MainWindow(interp);

/* Create commands & widgets */
Tcl_CreateCommand(interp, TclCmdName, tclCommandCallback,
(ClientData)NULL, (Tcl_CmdDeleteProc*)NULL);

if (Tcl_EvalFile(interp, inTclFilename) != TCL_OK) {
fprintf(stderr, "Tcl_EvalFile: error occured.\n");
fprintf(stderr, " %s: %d: %s\n",
inTclFilename, interp->errorLine, interp->result);
exit(1);
}

if (argc == 1) {
Tcl_Eval(interp, "popUpOpenCmd");
} else {
if (sscanf(argv[1], "z39.50s://%[^/]/%s", addr, dbname) != 2) {
fprintf(stderr, "ZURL invalid: %s\n", argv[1]);
fprintf(stderr, "addr: %s, dbname: %s\n\n", addr, dbname);
fprintf(stderr, "\tUsage: kappa ZURL\n");
fprintf(stderr, "\t\t(ZURL : z39.50s://host:port/dbname)\n");

exit(1);
}
if (sscanf(addr, "%[^:]:%s", host, port) != 2) {
strcpy(host, addr);
strcpy(port, "210");
}
}

```

```

sprintf(buf, "Searchproc \"open\" \"tcp:%s:%s\"\\n", host, port);
    sprintf(buf, "%s Searchproc \"base\" \"%s\"\\n", buf, dbname);
    strcat(buf, "wm deiconify .\\n");
    strcat(buf, "focus .entEnter.inputentry\\n");
    sprintf(buf, "%s wm title . \"検索画面 <%s:%s>\"\\n",
        buf, host, dbname);
    Tcl_Eval(interp, buf);
}

/* Main Loop */
Tk_MainLoop();

Tcl_DeleteInterp(interp);
Tcl_Exit(0);

/*NOTREACHED*/
return 0;
}

int tclCommandCallback(ClientData cd, int argc, char *argv[])
{
    int wait;

    static struct {
        char *cmd;
        int (*fun)(char *arg);
        char *ad;
    } cmd[] = {
        {"open", cmd_open, "('tcp'|'osi')':'<tssel>'/'<host>[:'<port>"]},
        {"quit", cmd_quit, ""},
        {"find", cmd_find, "<query>"},
        {"base", cmd_base, "<base-name>"},
        {"show", cmd_show, "<rec#>['+'<#recs>['+'<setname>]]"},
        {"scan", cmd_scan, "<term>"},
        {"sort", cmd_sort, "<sortkey> <flag> <sortkey> <flag> ..."},
        {"sort+", cmd_sort_newset, "<sortkey> <flag> <sortkey> <flag> ..."},
        {"authentication", cmd_authentication, "<acctstring>"},
        {"lslb", cmd_lslb, "<largeSetLowerBound>"},
        {"ssub", cmd_ssub, "<smallSetUpperBound>"},
        {"mspn", cmd_mspn, "<mediumSetPresentNumber>"},
        {"status", cmd_status, ""},
        {"setnames", cmd_setnames, ""},
        {"cancel", cmd_cancel, ""},
        {"format", cmd_format, "<recordsyntax>"},
        {"elements", cmd_elements, "<elementSetName>"},
        {"close", cmd_close, ""},

```

```

{"attributeset", cmd_attributeset, "<attrset>"},
    {"querytype", cmd_querytype, "<type>"},
    {0,0}
};
char *netbuffer= 0;
int netbufferlen = 0;
int i;
Z_APDU *apdu;
int res;
char tcl_cmd[BUFSIZE];
char tcl_entry[BUFSIZE];

strcpy(tcl_cmd,Tcl_GetVar(interp,"TclComandName",0));
strcpy(tcl_entry,Tcl_GetVar(interp,"TclEntryArg",0));

printf("\n コマンド名 : %s\n", tcl_cmd);
printf("入力文字列 : %s\n", tcl_entry);

#ifdef USE_SELECT
    fd_set input;
#endif

#ifdef USE_SELECT
    FD_ZERO(&input);
    FD_SET(0, &input);
    if (conn)
FD_SET(cs_fileno(conn), &input);
    if ((res = select(20, &input, 0, 0, 0)) < 0) {
perror("select");
exit(1);
    }
    if (!res)
continue;
    if (!wait && FD_ISSET(0, &input))
#else
/* この */ wait = 0;
    if (!wait)
#endif
    {
for (i = 0; cmd[i].cmd; i++)
    if (!strcmp(cmd[i].cmd, tcl_cmd, strlen(tcl_cmd))) {
res = (*cmd[i].fun)(tcl_entry);
break;
    }
if (res < 2) {
return 0;

```

```

}
}
#ifdef USE_SELECT
    if (conn && FD_ISSET(cs_fileno(conn), &input))
#endif
{
do {
    if ((res = cs_get(conn, &netbuffer, &netbufferlen)) < 0) {
perror("cs_get");
exit(1);
    }
    if (!res) {
printf("Target closed connection.\n");
unlink(tcldumpfilename);
exit(1);
    }
    odr_reset(in); /* release APDU from last round */
    odr_setbuf(in, netbuffer, res, 0);
    if (!z_APDU(in, &apdu, 0)) {
odr_perror(in, "Decoding incoming APDU");
fprintf(stderr, "[Near %d]\n", odr_offset(in));
fprintf(stderr, "Packet dump:\n-----\n");
odr_dumpBER(stderr, netbuffer, res);
fprintf(stderr, "-----\n");
exit(1);
    }
}
#if 0
    if (!z_APDU(print, &apdu, 0)) {
odr_perror(print, "Failed to print incoming APDU");
odr_reset(print);
continue;
    }
#endif
    switch(apdu->which) {
        case Z_APDU_initResponse:
process_initResponse(apdu->u.initResponse);
break;
        case Z_APDU_searchResponse:
process_searchResponse(apdu->u.searchResponse);
break;
        case Z_APDU_scanResponse:
process_scanResponse(apdu->u.scanResponse);
break;
        case Z_APDU_presentResponse:
setno += *apdu->u.presentResponse->numberOfRecordsReturned;
if (apdu->u.presentResponse->records)

```

```

        display_records(apdu->u.presentResponse->records);
else
    printf("No records.\n");
break;
    case Z_APDU_sortResponse:
process_sortResponse(apdu->u.sortResponse);
break;
    case Z_APDU_close:
printf("Target has closed the association.\n");
process_close(apdu->u.close);
break;
    default:
printf("Received unknown APDU type (%d).\n",
        apdu->which);
exit(1);
    }
    /* printf("setNo[%d]>>",setnumber + 1);
    fflush(stdout); */
} while (cs_more(conn));
    }
    wait = 0;
    return TCL_OK;
}

int main(int argc, char **argv)
{
    initialize();
    sprintf(tclDumpfilename, "%s/%s%d", TMPDIR, PACKAGE, getpid());
    cmd_base("Default");
    printf("setNo[%d]>>",setnumber + 1);

    return client(argc, argv);
}

void tclPrintf(char *str, int lnum)
{
    char buf[BUFSIZE];

    /* Tcl_SetVar(interp,"setdata", str,0); */
    if (lnum == 2) {
sprintf(buf, "errWin {%s}", str);
    } else if(lnum == 4){
sprintf(buf, "errWin_open {%s}", str);
    } else{
        printf(" tclPrintf() error lnum \n");
        exit(1);
    }
}

```



```
}  
Tcl_Eval(interp, buf);  
}
```

D.2 client.tcl

```

# -*- Mode: Tcl -*-
#
# Kappa : Japanese Z39.50(-1995) Client.
#
# Copyright (C) 1997-1998 Yuka Egusa. All rights reserved.
# Copyright (C) 1999 Yuka Egusa and Masao Takaku. All rights reserved.

global version
set version "0.11"

global conf

# 色の指定
set conf(fgcolor) {black}
set conf(bgcolor) {#c0dfff}
set conf(hlcolor) {blue} ;# 強調色

# (デモ用に) ウィンドウのサイズ・位置を固定する。
set conf(openGeometry) 633x518+220+120
set conf(searchGeometry) 681x607+200+100

# データファイルの置場所
set conf(datadir) [file dirname [info script]]

. configure -bg $conf(bgcolor)

# C 言語と、コマンドをやりとりしている関数 -----
proc Searchproc {cmd entry} {
    #puts $cmd
    #puts $entry
    set TclComandName $cmd
    set TclEntryArg $entry
    TclClientCommand
}

# 指定されたファイルをオープンして、指定されたリストボックスに書き込む関数---
proc recordToListbox {filename listboxname} {
    set fd [open $filename r]
    while { [gets $fd line] >= 0 } { $listboxname insert end $line }
    close $fd
}

# 著作権表示を行なう。

```

```

proc copyright {} {
    global conf version
    toplevel .about
    .about configure -bg $conf(bgcolor)

    set copyright "Kappa : Japanese Z39.50 Client v$version\n"
    append copyright "Copyright (C) 1999\n"
    append copyright " Yuka Egusa <yuka@ulis.ac.jp>\n"
    append copyright " Masao Takaku <masao@ulis.ac.jp>\n"
    append copyright "All right reserved.\n"
    message .about.textmessage -fg $conf(fgcolor) -bg $conf(bgcolor) \
    -width 700 -padx 1c -pady 1c -text $copyright

    button .about.fin -bg $conf(bgcolor) -fg $conf(fgcolor) \
    -text " Close Help " -command {destroy .about}

    pack .about.textmessage .about.fin
    wm title .about "About Kappa"
}

# エラーメッセージを出して オープンウィンドウを出す 関数
proc errWin_open {str} {
    global conf
    toplevel .err1

    frame .err1.f -bg $conf(bgcolor)
    label .err1.f.text -bg $conf(bgcolor) -fg $conf(fgcolor) -text $str -relief ridge
    label .err1.f.errorimage -bg $conf(bgcolor) -fg $conf(fgcolor) -bitmap error
    pack .err1.f.text .err1.f.errorimage -in .err1.f -side left

    button .err1.fin -fg $conf(fgcolor) -bg $conf(bgcolor) -text " OK ? " -command {
        destroy .err1
    }
    #wm deiconify .topen
    exit 1
    #popUpOpenCmd
}
pack .err1.f .err1.fin
wm title .err "接続エラー"
}

# エラーメッセージを出す 関数
proc errWin {str} {
    global conf
    toplevel .err1

    frame .err1.f -bg $conf(bgcolor)

```

```

label .err1.f.text -bg $conf(bgcolor) -fg $conf(fgcolor) -text $str -relief ridge
label .err1.f.errorimage -bg $conf(bgcolor) -fg $conf(fgcolor) -bitmap error
pack .err1.f.text .err1.f.errorimage -in .err1.f -side left

button .err1.fin -bg $conf(bgcolor) -fg $conf(fgcolor) -text " OK ? " \
-command { destroy .err1 }
pack .err1.f .err1.fin
wm title .err "エラー"
}

proc inputShowEntry {setnum beginnum endnum} {
    .showEnter.setno delete 0 end
    .showEnter.beginno delete 0 end
    .showEnter.endno delete 0 end
    .showEnter.setno insert 0 $setnum
    .showEnter.beginno insert 0 $beginnum
    .showEnter.endno insert 0 $endnum
}

#proc urlExtract {str host port dbname} {
#    regexp {^z39.50s://([^/:]*)?:?([0-9]*)/(.*)$} $str match \
#    host port dbname
#}

# ホスト名などの情報を取得する。
proc getDBInfo {filename} {
    global host port dbname format comment
    set host(default) ""
    set port(default) "210"
    set dbname(default) "Default"
    set format(default) "USMARC"
    set comment(default) ""

    set infile [ open "$filename" r ]
    set id 0
    while { [ gets $infile line ] >= 0 } {
if { [ regexp {^z39.50s://([^/:]*)?:?([0-9]*)/([^#]*)#([^#]*)#?(.*)?$} $line match host($id)
port($id) dbname($id) comment($id) format($id) ]} {
    if { ! [ string compare $port($id) "" ] } {
set port($id) $port(default)
    }
    #puts "$host($id) $port($id) $dbname($id)"
} else {
    continue
}
incr id

```

```

}
close $infile
# puts [ parray host ] [ parray port ]
}

proc inputOpenEntry {host port DBname} {
    .topen.e0 delete 0 end
    .topen.e1 delete 0 end
    .topen.e2 delete 0 end
    .topen.e0 insert 0 $host
    .topen.e1 insert 0 $port
    .topen.e2 insert 0 $DBname
}

# menubutton を作る。
# 第 1 引数は名前、第 2 引数は親 Widget のパス名をとる。
proc makeMenuButton {name parent} {
    global conf
    # 最初の 1 文字は大文字にする。
    set first [string toupper [string range $name 0 0]]
    set remain [string range $name 1 [string length $name]]
    append label $first $remain
    # puts "ラベル: $label"

    menubutton $parent.$name -ind true -relief flat -under 0 \
    -indicatoron false -fg $conf(fgcolor) -bg $conf(bgcolor) \
    -text $label -menu $parent.$name.menu
    # -font $fontsize
}

# 接続ウィンドウの定義
proc popUpOpenCmd {} {
    global conf host port dbname format comment

    toplevel .topen
    wm withdraw .
    .topen configure -bg $conf(bgcolor)

    # メニューバーの定義
    frame .topen.menuubar -bg $conf(bgcolor)
    makeMenuButton file .topen.menuubar
    makeMenuButton help .topen.menuubar

    menu .topen.menuubar.file.menu -fg $conf(fgcolor) -bg $conf(bgcolor)
    # -font $fontsize
    .topen.menuubar.file.menu add command -label "Exit" -command {

```

```

destroy .
}
menu .topen.menubar.help.menu -fg $conf(fgcolor) -bg $conf(bgcolor)
# -font $fontsize
.topen.menubar.help.menu add command -label "About Kappa" -command {
copyright
}
pack .topen.menubar.file -side left
pack .topen.menubar.help -side right

label .topen.l0 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-relief groove -text "Host"
label .topen.l1 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-relief groove -text "Port"
label .topen.l2 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-relief groove -text "DataBase Name"

entry .topen.e0 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-width 24 -relief sunken -textvariable openhostdata
entry .topen.e1 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-width 7 -relief sunken -textvariable openportdata
entry .topen.e2 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-width 16 -relief sunken -textvariable openbasedata

button .topen.button -fg $conf(fgcolor) -bg $conf(bgcolor) \
-bg $conf(hlcolor) -fg white -text "connect" -command {
set openentdata "tcp:${openhostdata}:${openportdata}"

destroy .topen
#wm withdraw .topen
wm title . "検索画面 <${openhostdata}:${openbasedata}>"
Searchproc "open" $openentdata

Searchproc "base" $openbasedata
wm deiconify .
focus .entEnter.inputentry
}

# ----- リストボックスを作っている部分 (2つ) -----
listbox .topen.li0 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-relief groove -width 60 -height 17 \
-yscrollcommand ".topen.s1 set" -xscrollcommand ".topen.s2 set"

# set basedir [ file dirname [ info script ] ]
getDBInfo "$conf(datadir)/opensite"
# puts [parray comment]

```

```

    set size [ array size comment ]
    set i "0"
    while { $i < $size-1 } {
.topen.li0 insert end "$comment($i)"
incr i
    }

    bind .topen.li0 <ButtonRelease-1> {
set id [ .topen.li0 curselection ]
inputOpenEntry "$host($id)" "$port($id)" "$dbname($id)"
if { [ string compare $format($id) "" ] } {
    Searchproc "format" $format($id)
    .showEnter.recordSyntax configure -text "$format($id)"
}
}

.topen.li0 insert 0 "aladdin (zwais)"
.topen.li0 insert 1 "AGRICOLA"
.topen.li0 insert 2 "dranet"
.topen.li0 insert 3 "Univercity of Wisconsin-madison"

scrollbar .topen.s1 -bg $conf(bgcolor) \
-orient vertical -command ".topen.li0 yview"
scrollbar .topen.s2 -bg $conf(bgcolor) \
-orient horizontal -command ".topen.li0 xview"
# -orient スクロールバーの向き vertical 縦 horizontal 横

message .topen.m -fg $conf(fgcolor) -bg $conf(bgcolor) -relief sunken -text ""

grid .topen.menubar -row 0 -column 0 -columnspan 4 -sticky ew

grid .topen.l0 -row 1 -column 1 -sticky ew
grid .topen.l1 -row 1 -column 2 -sticky ew
grid .topen.l2 -row 1 -column 3 -sticky ew

grid .topen.button -row 2 -column 0 -sticky ew
grid .topen.e0 -row 2 -column 1 -sticky ew
grid .topen.e1 -row 2 -column 2 -sticky ew
grid .topen.e2 -row 2 -column 3 -sticky ew

grid .topen.li0 -row 3 -column 0 -columnspan 4 -sticky nsew
grid .topen.s1 -row 3 -column 4 -sticky ns

grid .topen.s2 -row 4 -column 0 -columnspan 4 -sticky ew

#リストボックスのある行に

```

```

# ( ウィンドウを広げた時にリストボックスの大きさだけを大きくするため )
grid rowconfigure .topen 3 -weight 100
# リストボックスのある列に ( 上の行と同理由 )
grid columnconfigure .topen 0 -weight 100
grid columnconfigure .topen 1 -weight 100
grid columnconfigure .topen 2 -weight 100
grid columnconfigure .topen 3 -weight 100

.topen configure -bd 2
wm title .topen "接続画面"
wm geometry .topen $conf(openGeometry)
}

# ----- entry フレームを宣言しているところ -----
#set entdata ""
frame .entEnter -bg $conf(bgcolor)
label .entEnter.inputlabel -fg $conf(fgcolor) -bg $conf(bgcolor) -text " Query:"
#.entEnter.inputlabel conf -font "-*-*-*-*-20-*"
entry .entEnter.inputentry -fg white -bg $conf(hlcolor) \
-width 40 -relief sunken -textvariable entdata
#.entEnter.inputentry conf -font "-*-*-*-*-20-*"
#-fg $conf(fgcolor) -bg $conf(bgcolor)
button .entEnter.clear -fg $conf(fgcolor) -bg $conf(bgcolor) -text clear -command {
.entEnter.inputentry delete 0 end
}
pack .entEnter.inputlabel .entEnter.inputentry .entEnter.clear -in .entEnter -side left

frame .showEnter -bg $conf(bgcolor)
label .showEnter.label1 -fg $conf(fgcolor) -bg $conf(bgcolor) -text "\["
label .showEnter.label2 -fg $conf(fgcolor) -bg $conf(bgcolor) -text "\] No"
label .showEnter.label3 -fg $conf(fgcolor) -bg $conf(bgcolor) -text "~ No"
label .showEnter.label4 -fg $conf(fgcolor) -bg $conf(bgcolor) -text " "
entry .showEnter.setno -fg $conf(fgcolor) -bg $conf(bgcolor) \
-width 4 -relief sunken -textvariable setnodata
entry .showEnter.beginno -fg $conf(fgcolor) -bg $conf(bgcolor) \
-width 6 -relief sunken -textvariable beginnodata
entry .showEnter.endno -fg $conf(fgcolor) -bg $conf(bgcolor) \
-width 6 -relief sunken -textvariable endnodata

menubutton .showEnter.recordSyntax -ind true -relief raised\
-text "USMARC" -menu .showEnter.recordSyntax.menu -fg $conf(fgcolor) -bg $conf(bgcolor)
menu .showEnter.recordSyntax.menu -fg $conf(fgcolor) -bg $conf(bgcolor)
.showEnter.recordSyntax.menu add command -label "USMARC" -command {
.showEnter.recordSyntax configure -text "USMARC"
Searchproc "format" "usmarc"
}

```



```

.showEnter.recordSyntax.menu add command -label "SUTRS" -command {
    .showEnter.recordSyntax configure -text "SUTRS"
    Searchproc "format" "sutrs"
}
.showEnter.recordSyntax.menu add command -label "DANMARC" -command {
    .showEnter.recordSyntax configure -text "DANMARC"
    Searchproc "format" "danmarc"
}
.showEnter.recordSyntax.menu add command -label "UKMARC" -command {
    .showEnter.recordSyntax configure -text "UKMARC"
    Searchproc "format" "ukmarc"
}
.showEnter.recordSyntax.menu add command -label "GRS1" -command {
    .showEnter.recordSyntax configure -text "GRS1"
    Searchproc "format" "grs1"
}
.showEnter.recordSyntax.menu add command -label "SOIF" -command {
    .showEnter.recordSyntax configure -text "SOIF"
    Searchproc "format" "soif"
}
.showEnter.recordSyntax.menu add command -label "Summary" -command {
    .showEnter.recordSyntax configure -text "Summary"
    Searchproc "format" "summary"
}
.showEnter.recordSyntax.menu add command -label "Explain" -command {
    .showEnter.recordSyntax configure -text "Explain"
    Searchproc "format" "explain"
}
set element F
radiobutton .showEnter.full -text "Full" -value F -variable element \
-bg $conf(bgcolor)
radiobutton .showEnter.burief -text "Brief" -value B -variable element \
-bg $conf(bgcolor)
pack .showEnter.label1 .showEnter.setno .showEnter.label2 \
.showEnter.beginno .showEnter.label3 .showEnter.endno \
.showEnter.label4 \
.showEnter.full .showEnter.burief .showEnter.recordSyntax -side left

# ----- entry フレームを宣言 終了-----

# ----- ボタンを宣言しているところ -----
button .find -fg $conf(fgcolor) -bg $conf(bgcolor) -text Search -command {
    .find configure -fg $conf(fgcolor) -bg $conf(bgcolor)
    .entEnter.inputentry configure -fg $conf(fgcolor) -bg $conf(bgcolor)
    .show configure -fg white -bg $conf(hlcolor)
    set lastline0 [.10 index end]
}

```

```

Searchproc "find" $entdata
.l10 yview $lastline0
#Searchproc "find" [toYazSyntax $entdata]
#.entEnter.inputentry delete 0 end
}

bind .entEnter.inputentry <KeyPress> {
.find configure -fg white -bg $conf(hlcolor)
.entEnter.inputentry configure -fg white -bg $conf(hlcolor)
.show configure -fg $conf(fgcolor) -bg $conf(bgcolor)
}

bind .entEnter.inputentry <Key-Return> {
.find configure -fg $conf(fgcolor) -bg $conf(bgcolor)
.entEnter.inputentry configure -fg $conf(fgcolor) -bg $conf(bgcolor)
.show configure -fg white -bg $conf(hlcolor)
set lastline0 [.l10 index end]
Searchproc "find" $entdata
#.entEnter.inputentry delete 0 end
.l10 yview $lastline0
}

#button .show -fg $conf(fgcolor) -bg $conf(bgcolor) -text show -command {popUpShowCmd}
button .show -fg $conf(fgcolor) -bg $conf(bgcolor) -text show -command {
.show configure -fg $conf(fgcolor) -bg $conf(bgcolor)
.find configure -fg $conf(fgcolor) -bg $conf(bgcolor)
.entEnter.inputentry configure -fg white -bg $conf(hlcolor)
set lastline1 [.l11 index end]
set count [expr ${endnodata}-${beginnodata}+1]
Searchproc "elements" $element
Searchproc "show" "${beginnodata}+${count}@${setnodata}"
.l11 yview $lastline1
}

#メニューバーの定義
frame .menubar -bg $conf(bgcolor)
makeMenuButton file .menubar
makeMenuButton connect .menubar
makeMenuButton help .menubar

pack .menubar.file .menubar.connect -side left
pack .menubar.help -side right
# メニューバーの定義 ここまで

# ここから 各メニューの定義
# file メニューの宣言
menu .menubar.file.menu -fg $conf(fgcolor) -bg $conf(bgcolor)

```

```

# 以下 3 行のコメントアウトを外すと、
# 新しい接続ウィンドウが接続を切らずに出てくる
.menubar.file.menu add command -label "New Window" -command {
    exec kappa.bin &
}
.menubar.file.menu add command -label "Exit" -command {
    Searchproc "quit" ""
}

# connect メニュー
menu .menubar.connect.menu -fg $conf(fgcolor) -bg $conf(bgcolor)
.menubar.connect.menu add command -label "Disconnect" -command {
    Searchproc "close" ""
}

# help メニュー
menu .menubar.help.menu -fg $conf(fgcolor) -bg $conf(bgcolor)
.menubar.help.menu add command -label "About Query ..." -command {
    toplevel .helpquery

    .helpquery configure -bg $conf(bgcolor)
    message .helpquery.textmessage -bg $conf(bgcolor) -fg $conf(fgcolor) \
        -padx 1c -pady 1c -text {
「図書」を検索
図書

「書名にガラス」を検索
ti="ガラス"

「著者名に鈴木」を検索
au="鈴木"

1 番目と 2 番目の検索結果集合の AND 検索
set=1 and set=2

    }

    button .helpquery.fin -bg $conf(bgcolor) -fg $conf(fgcolor) \
        -text " Close " -command {destroy .helpquery}

    pack .helpquery.textmessage .helpquery.fin
}
.menubar.help.menu add command -label "About Kappa" -command {
    copyright
}
# -----ボタンの宣言 終了-----

```

```

# ----- リストボックスを作っている部分 (2つ) -----
listbox .l0 -bg $conf(bgcolor) -fg $conf(fgcolor) \
-relief groove -width 60 -height 5 \
-yscrollcommand ".s1 set" -xscrollcommand ".s2 set"
scrollbar .s1 -bg $conf(bgcolor) \
-orient vertical -command ".l0 yview"
scrollbar .s2 -bg $conf(bgcolor) \
-orient horizontal -command ".l0 xview"
# -orient スクロールバーの向き vertical 縦 horizontal 横

# リストで指定すると、指定したところの show の履歴検索ができる
# 指定したところ "@set number" が、検索エントリーに入力される (add 12/10)
bind .l0 <ButtonRelease-1> {
    # show エントリー用
    set setnum [expr [.l0 curselection] + 1]
    set liststr [.l0 get [.l0 curselection]]
    set len [expr [string length $liststr] - 1]
    set endnumber [string range $liststr [string wordstart $liststr $len] $len]
    if { $endnumber > 30} {
set endnumber 30
    }
    inputShowEntry $setnum 1 $endnumber

    # 背景色の変更
    .find configure -fg $conf(fgcolor) -bg $conf(bgcolor)
    .entEnter.inputentry configure -fg $conf(fgcolor) -bg $conf(bgcolor)
    .show configure -fg white -bg $conf(hlcolor)
}

bind .l0 <ButtonRelease-3> {
    set setnum [expr [.l0 curselection] + 1]
    # 検索エントリー用
    set buf " set="
    set queryterm $buf$setnum
    .entEnter.inputentry insert insert $queryterm

    # 背景色の変更
    .show configure -fg $conf(fgcolor) -bg $conf(bgcolor)
    .entEnter.inputentry configure -fg white -bg $conf(hlcolor)
    .find configure -fg white -bg $conf(hlcolor)
}

listbox .l1 -fg $conf(fgcolor) -bg $conf(bgcolor) \
-relief groove -width 60 -height 14 \
-yscrollcommand ".s3 set" -xscrollcommand ".s4 set"
scrollbar .s3 -bg $conf(bgcolor) \

```

```

-orient vertical -command ".l1 yview"
scrollbar .s4 -bg $conf(bgcolor) \
-orient horizontal -command ".l1 xview"

# ----- リストボックス 終了 -----

# ----- メッセージ -----
message .m -fg $conf(fgcolor) -bg $conf(bgcolor) -relief sunken -text ""
# ----- メッセージ 終了 -----
#----- 配置 -----
grid .menubar -row 0 -column 0 -columnspan 3 -sticky ew
grid .find -row 1 -column 0 -columnspan 2 -sticky ew
grid .entEnter -row 1 -column 2 -columnspan 2 -sticky ew
#grid .clear -row 1 -column 3 -columnspan 1 -sticky ew
grid .show -row 2 -column 0 -columnspan 2 -sticky ew
grid .showEnter -row 2 -column 2 -columnspan 2 -sticky ew

grid .l0 -row 3 -column 0 -columnspan 3 -sticky nsew
grid .s1 -row 3 -column 3 -sticky ns
grid .s2 -row 4 -column 0 -columnspan 3 -sticky ew
grid .l1 -row 5 -column 0 -columnspan 3 -sticky nsew
grid .s3 -row 5 -column 3 -sticky ns
grid .s4 -row 6 -column 0 -columnspan 3 -sticky ew

#リストボックスのある行に
#(ウィンドウを広げた時にリストボックスの大きさだけを大きくするため)
grid rowconfigure . 3 -weight 100
grid rowconfigure . 5 -weight 100
#リストボックスのある列に(上の行と同理由)
grid columnconfigure . 0 -weight 100
grid columnconfigure . 1 -weight 100
grid columnconfigure . 2 -weight 100

# popUpOpenCmd

# set mywidth [wininfo width .topen]
# set myheight [wininfo height .topen]
# puts [wininfo geometry .]
# puts [wininfo geometry .topen]
. configure -bd 2
wm geometry . $conf(searchGeometry)
#----- 配置終了 -----

```

D.3 Makefile

```
# Generated automatically from Makefile.in by configure.
# Makefile.in generated automatically by automake 1.4 from Makefile.am

# Copyright (C) 1994, 1995-8, 1999 Free Software Foundation, Inc.
# This Makefile.in is free software; the Free Software Foundation
# gives unlimited permission to copy and/or distribute it,
# with or without modifications, as long as this notice is preserved.

# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY, to the extent permitted by law; without
# even the implied warranty of MERCHANTABILITY or FITNESS FOR A
# PARTICULAR PURPOSE.

SHELL = /bin/sh

srcdir = .
top_srcdir = .
prefix = /home/yuka
exec_prefix = ${prefix}

bindir = ${exec_prefix}/bin
sbindir = ${exec_prefix}/sbin
libexecdir = ${exec_prefix}/libexec
datadir = ${prefix}/share
sysconfdir = ${prefix}/etc
sharedstatedir = ${prefix}/com
localstatedir = ${prefix}/var
libdir = ${exec_prefix}/lib
infodir = ${prefix}/info
mandir = ${prefix}/man
includedir = ${prefix}/include
oldincludedir = /usr/include

DESTDIR =

pkgdatadir = $(datadir)/kappa
pkglibdir = $(libdir)/kappa
pkgincludedir = $(includedir)/kappa

top_builddir = .

ACLOCAL = aclocal
```

```

AUTOCONF = autoconf
AUTOMAKE = automake
AUTOHEADER = autoheader

INSTALL = /usr/local/bin/install -c
INSTALL_PROGRAM = ${INSTALL} $(AM_INSTALL_PROGRAM_FLAGS)
INSTALL_DATA = ${INSTALL} -m 644
INSTALL_SCRIPT = ${INSTALL_PROGRAM}
transform = s,x,x,

NORMAL_INSTALL = :
PRE_INSTALL = :
POST_INSTALL = :
NORMAL_UNINSTALL = :
PRE_UNINSTALL = :
POST_UNINSTALL = :
CC = gcc
MAKEINFO = /home/yuka/kappadir/kappa/missing makeinfo
PACKAGE = kappa
RANLIB = :
VERSION = 0.11

AUTOMAKE_OPTIONS = no-dependencies

bin_SCRIPTS = kappa
bin_PROGRAMS = kappa.bin
kappa_bin_SOURCES = client.c

pkgdata_DATA = client.tcl opensite default.bib

EXTRA_DIST = kappa client.tcl.in opensite default.bib

# YAZ(Yet Another Toolkit) directory
YAZDIR = /home/masao/FreeSoftware/yaz-1.4pl2

# Tcl/Tk Includes & Libraries
TCLINC = -I/usr/local/include
TCLLIB = -L/usr/local/lib -ltcl8.0jp${TCL_DBGX} -lm

TKINC = # no special path needed
TKLIB = -L/usr/local/lib -ltk8.0jp${TK_DBGX} -L/usr/local/lib -ltcl8.0jp${TCL_DBGX} -lX11 -lm

# Compiler options
INCLUDE = -I$(YAZDIR)/include $(TCLINC) $(TKINC)
DEFS = -DCCL2RPN=1 $(INCLUDE)

```

```

kappa_bin_LDADD = $(YAZDIR)/lib/libyaz.a $(YAZDIR)/lib/ccl.a $(TCLLIB) $(TKLIB)
ACLOCAL_M4 = $(top_srcdir)/aclocal.m4
mkinstalldirs = $(SHELL) $(top_srcdir)/mkinstalldirs
CONFIG_HEADER = config.h
CONFIG_CLEAN_FILES = client.tcl
PROGRAMS = $(bin_PROGRAMS)

CPPFLAGS =
LDFLAGS =
LIBS =
kappa_bin_OBJECTS = client.o
kappa_bin_DEPENDENCIES = $(YAZDIR)/lib/libyaz.a $(YAZDIR)/lib/ccl.a
kappa_bin_LDFLAGS =
SCRIPTS = $(bin_SCRIPTS)

CFLAGS = -g -O2
COMPILE = $(CC) $(DEFS) $(INCLUDES) $(AM_CPPFLAGS) $(CPPFLAGS) $(AM_CFLAGS) $(CFLAGS)
CCLD = $(CC)
LINK = $(CCLD) $(AM_CFLAGS) $(CFLAGS) $(LDFLAGS) -o $@
DATA = $(pkgdata_DATA)

DIST_COMMON = README ./stamp-h.in AUTHORS COPYING ChangeLog INSTALL \
Makefile.am Makefile.in NEWS TODO acconfig.h aclocal.m4 client.tcl.in \
config.h.in configure configure.in install-sh missing mkinstalldirs

DISTFILES = $(DIST_COMMON) $(SOURCES) $(HEADERS) $(TEXINFOS) $(EXTRA_DIST)

TAR = tar
GZIP_ENV = --best
SOURCES = $(kappa_bin_SOURCES)
OBJECTS = $(kappa_bin_OBJECTS)

all: all-redirect
.SUFFIXES:
.SUFFIXES: .S .c .o .s
$(srcdir)/Makefile.in: Makefile.am $(top_srcdir)/configure.in $(ACLOCAL_M4)
cd $(top_srcdir) && $(AUTOMAKE) --gnu Makefile

Makefile: $(srcdir)/Makefile.in $(top_builddir)/config.status
cd $(top_builddir) \
  && CONFIG_FILES=$@ CONFIG_HEADERS= $(SHELL) ./config.status

$(ACLOCAL_M4): configure.in
cd $(srcdir) && $(ACLOCAL)

```



```

config.status: $(srcdir)/configure $(CONFIG_STATUS_DEPENDENCIES)
$(SHELL) ./config.status --recheck
$(srcdir)/configure: $(srcdir)/configure.in $(ACLOCAL_M4) $(CONFIGURE_DEPENDENCIES)
cd $(srcdir) && $(AUTOCONF)

```

```

config.h: stamp-h
@if test ! -f $@; then \
rm -f stamp-h; \
$(MAKE) stamp-h; \
else ;; fi
stamp-h: $(srcdir)/config.h.in $(top_builddir)/config.status
cd $(top_builddir) \
&& CONFIG_FILES= CONFIG_HEADERS=config.h \
$(SHELL) ./config.status
@echo timestamp > stamp-h 2> /dev/null
$(srcdir)/config.h.in: $(srcdir)/stamp-h.in
@if test ! -f $@; then \
rm -f $(srcdir)/stamp-h.in; \
$(MAKE) $(srcdir)/stamp-h.in; \
else ;; fi
$(srcdir)/stamp-h.in: $(top_srcdir)/configure.in $(ACLOCAL_M4) acconfig.h
cd $(top_srcdir) && $(AUTOHEADER)
@echo timestamp > $(srcdir)/stamp-h.in 2> /dev/null

```

```
mostlyclean-hdr:
```

```
clean-hdr:
```

```
distclean-hdr:
-rm -f config.h

```

```
maintainer-clean-hdr:
client.tcl: $(top_builddir)/config.status client.tcl.in
cd $(top_builddir) && CONFIG_FILES=$@ CONFIG_HEADERS= $(SHELL) ./config.status

```

```
mostlyclean-binPROGRAMS:
```

```
clean-binPROGRAMS:
-test -z "$(bin_PROGRAMS)" || rm -f $(bin_PROGRAMS)

```

```
distclean-binPROGRAMS:
```

```
maintainer-clean-binPROGRAMS:
```

```
install-binPROGRAMS: $(bin_PROGRAMS)

```

```

@$(NORMAL_INSTALL)
$(mkinstalldirs) $(DESTDIR)$(bindir)
@list='$(bin_PROGRAMS)'; for p in $$list; do \
  if test -f $$p; then \
    echo " $(INSTALL_PROGRAM) $$p $(DESTDIR)$(bindir)/'echo $$p|sed 's/$(EXEEXT)$$//'|sed
'$(transform)'|sed 's/$$/$(EXEEXT)/'"; \
    $(INSTALL_PROGRAM) $$p $(DESTDIR)$(bindir)/'echo $$p|sed 's/$(EXEEXT)$$//'|sed '$(transform)'|sed
's/$$/$(EXEEXT)/'"; \
  else ;; fi; \
done

uninstall-binPROGRAMS:
@$(NORMAL_UNINSTALL)
list='$(bin_PROGRAMS)'; for p in $$list; do \
  rm -f $(DESTDIR)$(bindir)/'echo $$p|sed 's/$(EXEEXT)$$//'|sed '$(transform)'|sed 's/$$/$(EXEEXT)/'"; \
done

.c.o:
$(COMPILE) -c $<

.s.o:
$(COMPILE) -c $<

.S.o:
$(COMPILE) -c $<

mostlyclean-compile:
-rm -f *.o core *.core

clean-compile:

distclean-compile:
-rm -f *.tab.c

maintainer-clean-compile:

kappa.bin: $(kappa_bin_OBJECTS) $(kappa_bin_DEPENDENCIES)
@rm -f kappa.bin
$(LINK) $(kappa_bin_LDFLAGS) $(kappa_bin_OBJECTS) $(kappa_bin_LDADD) $(LIBS)

install-binSCRIPTS: $(bin_SCRIPTS)
@$(NORMAL_INSTALL)
$(mkinstalldirs) $(DESTDIR)$(bindir)
@list='$(bin_SCRIPTS)'; for p in $$list; do \
  if test -f $$p; then \

```

```

    echo " $(INSTALL_SCRIPT) $$p $(DESTDIR)$(bindir)/'echo $$p|sed '$(transform)''"; \
    $(INSTALL_SCRIPT) $$p $(DESTDIR)$(bindir)/'echo $$p|sed '$(transform)''; \
else if test -f $(srcdir)/$$p; then \
    echo " $(INSTALL_SCRIPT) $(srcdir)/$$p $(DESTDIR)$(bindir)/'echo $$p|sed '$(transform)''"; \
\
    $(INSTALL_SCRIPT) $(srcdir)/$$p $(DESTDIR)$(bindir)/'echo $$p|sed '$(transform)''; \
    else ;; fi; fi; \
done

uninstall-binSCRIPTS:
@$(NORMAL_UNINSTALL)
list='$(bin_SCRIPTS)'; for p in $$list; do \
    rm -f $(DESTDIR)$(bindir)/'echo $$p|sed '$(transform)''; \
done

install-pkgdataDATA: $(pkgdata_DATA)
@$(NORMAL_INSTALL)
$(mkinstalldirs) $(DESTDIR)$ (pkgdatadir)
@list='$(pkgdata_DATA)'; for p in $$list; do \
    if test -f $(srcdir)/$$p; then \
        echo " $(INSTALL_DATA) $(srcdir)/$$p $(DESTDIR)$ (pkgdatadir)/$$p"; \
        $(INSTALL_DATA) $(srcdir)/$$p $(DESTDIR)$ (pkgdatadir)/$$p; \
    else if test -f $$p; then \
        echo " $(INSTALL_DATA) $$p $(DESTDIR)$ (pkgdatadir)/$$p"; \
        $(INSTALL_DATA) $$p $(DESTDIR)$ (pkgdatadir)/$$p; \
    fi; fi; \
done

uninstall-pkgdataDATA:
@$(NORMAL_UNINSTALL)
list='$(pkgdata_DATA)'; for p in $$list; do \
    rm -f $(DESTDIR)$ (pkgdatadir)/$$p; \
done

tags: TAGS

ID: $(HEADERS) $(SOURCES) $(LISP)
list='$(SOURCES) $(HEADERS)'; \
unique='for i in $$list; do echo $$i; done | \
    awk '    { files[$$0] = 1; } \
    END { for (i in files) print i; }''; \
here='pwd' && cd $(srcdir) \
    && mkid -f$$here/ID $$unique $(LISP)

TAGS: $(HEADERS) $(SOURCES) config.h.in $(TAGS_DEPENDENCIES) $(LISP)
tags=; \

```

```

here='pwd'; \
list='$(SOURCES) $(HEADERS)'; \
unique='for i in $$list; do echo $$i; done | \
  awk ' { files[$$0] = 1; } \
  END { for (i in files) print i; }'; \
test -z "$(ETAGS_ARGS)config.h.in$$unique$(LISP)$$tags" \
  || (cd $(srcdir) && etags $(ETAGS_ARGS) $$tags config.h.in $$unique $(LISP) -o $$here/TAGS)

mostlyclean-tags:

clean-tags:

distclean-tags:
-rm -f TAGS ID

maintainer-clean-tags:

distdir = $(PACKAGE)-$(VERSION)
top_distdir = $(distdir)

# This target untars the dist file and tries a VPATH configuration. Then
# it guarantees that the distribution is self-contained by making another
# tarfile.
distcheck: dist
-rm -rf $(distdir)
GZIP=$(GZIP_ENV) $(TAR) xzf $(distdir).tar.gz
mkdir $(distdir)/=build
mkdir $(distdir)/=inst
dc_install_base='cd $(distdir)/=inst && pwd'; \
cd $(distdir)/=build \
  && ./configure --srcdir=.. --prefix=$$dc_install_base \
  && $(MAKE) $(AM_MAKEFLAGS) \
  && $(MAKE) $(AM_MAKEFLAGS) dvi \
  && $(MAKE) $(AM_MAKEFLAGS) check \
  && $(MAKE) $(AM_MAKEFLAGS) install \
  && $(MAKE) $(AM_MAKEFLAGS) installcheck \
  && $(MAKE) $(AM_MAKEFLAGS) dist
-rm -rf $(distdir)
@banner="$(distdir).tar.gz is ready for distribution"; \
dashes='echo "$$banner" | sed s/./=/g'; \
echo "$$dashes"; \
echo "$$banner"; \
echo "$$dashes"
dist: distdir
-chmod -R a+r $(distdir)
GZIP=$(GZIP_ENV) $(TAR) chof $(distdir).tar.gz $(distdir)

```

```

-rm -rf $(distdir)
dist-all: distdir
-chmod -R a+r $(distdir)
GZIP=$(GZIP_ENV) $(TAR) chozf $(distdir).tar.gz $(distdir)
-rm -rf $(distdir)
distdir: $(DISTFILES)
-rm -rf $(distdir)
mkdir $(distdir)
-chmod 777 $(distdir)
@for file in $(DISTFILES); do \
  d=$(srcdir); \
  if test -d $$d/$$file; then \
    cp -pr $$d/$$file $(distdir)/$$file; \
  else \
    test -f $(distdir)/$$file \
    || ln $$d/$$file $(distdir)/$$file 2> /dev/null \
    || cp -p $$d/$$file $(distdir)/$$file || :; \
  fi; \
done
info-am:
info: info-am
dvi-am:
dvi: dvi-am
check-am: all-am
check: check-am
installcheck-am:
installcheck: installcheck-am
all-recursive-am: config.h
$(MAKE) $(AM_MAKEFLAGS) all-recursive

install-exec-am: install-binPROGRAMS install-binSCRIPTS
install-exec: install-exec-am

install-data-am: install-pkgdataDATA
install-data: install-data-am

install-am: all-am
@$(MAKE) $(AM_MAKEFLAGS) install-exec-am install-data-am
install: install-am
uninstall-am: uninstall-binPROGRAMS uninstall-binSCRIPTS \
uninstall-pkgdataDATA
uninstall: uninstall-am
all-am: Makefile $(PROGRAMS) $(SCRIPTS) $(DATA) config.h
all-redirect: all-am
install-strip:
$(MAKE) $(AM_MAKEFLAGS) AM_INSTALL_PROGRAM_FLAGS=-s install

```

```

installdirs:
$(mkinstalldirs) $(DESTDIR)$(bindir) $(DESTDIR)$(bindir) \
$(DESTDIR)$(pkgdatadir)

mostlyclean-generic:

clean-generic:

distclean-generic:
-rm -f Makefile $(CONFIG_CLEAN_FILES)
-rm -f config.cache config.log stamp-h stamp-h[0-9]*

maintainer-clean-generic:
mostlyclean-am: mostlyclean-hdr mostlyclean-binPROGRAMS \
mostlyclean-compile mostlyclean-tags \
mostlyclean-generic

mostlyclean: mostlyclean-am

clean-am: clean-hdr clean-binPROGRAMS clean-compile clean-tags \
clean-generic mostlyclean-am

clean: clean-am

distclean-am: distclean-hdr distclean-binPROGRAMS distclean-compile \
distclean-tags distclean-generic clean-am

distclean: distclean-am
-rm -f config.status

maintainer-clean-am: maintainer-clean-hdr maintainer-clean-binPROGRAMS \
maintainer-clean-compile maintainer-clean-tags \
maintainer-clean-generic distclean-am
@echo "This command is intended for maintainers to use;"
@echo "it deletes files that may require special tools to rebuild."

maintainer-clean: maintainer-clean-am
-rm -f config.status

.PHONY: mostlyclean-hdr distclean-hdr clean-hdr maintainer-clean-hdr \
mostlyclean-binPROGRAMS distclean-binPROGRAMS clean-binPROGRAMS \
maintainer-clean-binPROGRAMS uninstall-binPROGRAMS install-binPROGRAMS \
mostlyclean-compile distclean-compile clean-compile \
maintainer-clean-compile uninstall-binSCRIPTS install-binSCRIPTS \
uninstall-pkgdataDATA install-pkgdataDATA tags mostlyclean-tags \

```

```
distclean-tags clean-tags maintainer-clean-tags distdir info-am info \  
dvi-am dvi check check-am installcheck-am installcheck all-recursive-am \  
install-exec-am install-exec install-data-am install-data install-am \  
install uninstall-am uninstall all-redirect all-am all installdirs \  
mostlyclean-generic distclean-generic clean-generic \  
maintainer-clean-generic clean mostlyclean distclean maintainer-clean
```

```
# Tell versions [3.59,3.63) of GNU make to not export all variables.  
# Otherwise a system limit (for SysV at least) may be exceeded.  
.NOEXPORT:
```

D.4 kappa

```
#!/bin/sh

client="'dirname $0'/kappa.bin"

if [ -z "$DISPLAY" ]; then
    echo "Error: can't open display."
    echo "Please set DISPLAY variable"
    exit 1
fi

if [ ! -x "$client" ]; then
    echo "Error: can't execute $client."
    exit 1
fi

$client $*
while test $? -ne 0; do
    $client
done
exit 0

# while true; do
#     $client $*
#     if [ $? -eq 0 ]; then
#         exit 0
#     fi
# done
```