論　文

スクリプトに基づくWWW情報統合支援システムとゲノムデータベースへの応用

北村　泰彦†　　野崎　哲也†　　辰巳　昭治†

A Script-Based WWW Information Integration Support System and Its Application to Genome Databases

Yasuhiko KITAMURA†, Tetsuya NOZAKI†, and Shoji TATSUMI†

あらまし　インターネットの普及と共にWWWは情報発信の手段として最も広く利用されている技術の一つである．しかしながらWWWの情報源は広く分散し，個別に管理されているので，WWWブラウザを介して情報統合を行うためには利用者の側で多くの作業を行わなければならない．MetaCommanderは利用者がその要求をスクリプト言語により記述することで，WWW上での情報収集や統合を代行してくれるシステムである．本論文ではMetaCommanderのシステム構成とスクリプト言語について述べた後，WWWを介して公開されているゲノムデータベースへの応用について述べ，その有効性を示す．更に既存のスクリプト言語やプログラミング言語との比較を行い，エージェント技術の導入による今後の展望を示す．

キーワード　情報統合，WWW，エージェント言語，ゲノム情報学

1. まえがき

インターネットはその急速な普及に象徴されるように，いまや現代社会をささえるインフラストラクチャの一つとなりつつある．その中でもWWW（World Wide Web）は情報発信の手段として最も広く利用されている技術の一つであるが，その原因としては，利用できるメディアの多様さ，レイアウトに関する自由度の大きさ，情報発信の容易さ，情報の関連性を示すハイパリンクなどが挙げられる．

このような利点は情報発信者に対しては大きな利益をもたらすが，情報受信者の側では以下のような問題が生じている．

（1）　必要な情報が断片的に膨大な数のサーバに分散して存在しており，それらの情報を収集することが容易でない．

（2）　ブラウザを用いて多量の情報にアクセスしようとする場合には，繰り返し作業が多くなり，操作がわずらわしい．

（3）　さまざまな情報が一つのページに混在している場合があり，構造化がなされていない．HTML（Hyper Text Markup Language）[7]は視覚的な構造を定義できるが，意味的な構造を反映しているわけではない．従って機械的な情報の解析は困難である．

これらの問題点は分散している情報をまとめ上げ，付加価値の高い情報を提供するという情報統合（Information Integration）[12]を困難にしている．そこでこれらの問題点に対処するために，これまでにさまざまなツールやシステムが提案されている．

（1）の問題に対処する代表的なシステムは検索エンジンである．検索エンジンにはYahoo!(注1)などのように情報提供者からの登録に基づくものと，AltaVista(注2)なのどようにロボットによる情報収集に基づくものがある．前者の方は情報がカテゴリーごとに整理されているが，検索可能な情報量は後者の方が多いという長所短所がある．情報検索の手段としてはキーワード検索を用いることが一般的であるが，所望の情報以外のものが多くヒットし，その精度は必ずしも高くない．

† 大阪市立大学工学部情報工学科，大阪市
Department of Information and Communication Engineering, Faculty of Engineering, Osaka City University, 3-3-138 Sugimoto, Sumiyoshi-ku, Osaka-shi, 558-8585 Japan

(注1)：http://www.yahoo.com
(注2)：http://www.altavista.com

（2）の問題に対処するツールとしてはオートパイロットプログラムがある．オートパイロットプログラムは利用者が指定した階層だけ自動的にリンクをたどり，サーバからページを収集し，ファイル化してくれる．これは頻繁に更新されるページを自動的に収集し，オフラインでブラウジングをするような場合には有用である．オートパイロットプログラムは LiveAgent(注3) [6] のように単独で商品化されているものもあるが，最近ではブラウザ自体に組み込まれつつある．しかし現在のオートパイロットプログラムの機能は限定されており，ブラウザ上での複雑な操作を自動化できるわけではない．

（3）の問題に対してはデータベースの分野において，WWW 情報を一般のデータベースに統合化する試みがなされている．TSIMMIS プロジェクト [4] においては変換器を用いて WWW 情報の構造化を行い，オブジェクト指向データベースへと統合化している．しかし情報構造の異なるページごとに変換器を用意する必要があり，不特定多数の WWW ページを容易に統合できるわけではない．

このように以上のいずれのアプローチにしても汎用的なシステムを目指す場合には，実現できる機能は限定されたものとなるというトレードオフの問題が生じる．逆に対象とする領域を限定し，質の高いサービスを提供しようとする試みとして Internet Softbot [3] の研究がある．Softbot の代表例には複数の代表的な検索エンジンを統合することで質の高い情報検索サービスを提供する MetaCrawler [9](注4)，名前や所属から個人のホームページを検索する Ahoy! [10](注5)，電子モールでの比較ショッピングのための ShopBot [2](注6) などが挙げられる．

Internet Softbot は WWW サーバのような 1 次情報源から情報を収集，統合し，質の高い情報を不特定多数の利用に提供する 2 次情報源とみなすことができる．一方で，利用者は 1 次情報源から得られる情報を個人的にカスタマイズし，統合化したいという要求をもっているが，その要求にこたえるためには情報統合を第三者が 2 次情報源のサーバ上で行うのではなく，利用者自らのクライアント上で行えることが望ましい．すなわちクライアントとなるワークステーションあるいはパソコン上で利用者の要求を満たすような情報統合を容易に行えることが重要である．

そこで我々はクライアント側での WWW 情報統合を支援する MetaCommander を開発した．MetaCommander では利用者の要求はスクリプトとして記述され，その機能には WWW 情報の検索，抽出，整形と共に，ファイルアクセス，変数操作，数値/論理演算，繰返し操作などが挙げられる．本論文では MetaCommander をゲノムデータベースの分野に応用することにより，その有効性を示した．また従来のスクリプト言語やプログラミング言語との比較，また機能を更に高度化させるために必要なエージェント技術との関連について考察を加えている．

2. MetaCommander

2.1 設計方針

MetaCommander は以下のような方針に基づいて設計がなされている．

（A） 分散している WWW サーバからの情報統合に対する利用者の要求は簡易スクリプト言語により表す．これにより機能は限定されるものの，Internet Softbot のように高級言語で記述するよりも，容易に情報統合を行うことができる．また計算機の専門家でなくても記述が可能になる．

（B） 情報統合の単位はページよりも細かくする．これにより複数の情報が混在するようなページから必要な情報を切り出すことができる．

（C） より高度な情報統合を実現するために，通常の計算機言語がもつようなファイルアクセス，数値/論理計算，制御機能をもたせる．

以上の特徴により，MetaCommander は **1.** で述べた従来の情報統合における問題点に以下のように対処している．(1) に関しては分散している WWW サーバの URL をスクリプト中に組み込むことにより対処する．これによりスクリプトの実行により分散している情報を自動的に統合できる．但しこの方法では新たなサーバが加わったり，変更される場合には十分に対処できない．これに対しては **4.** で述べるように，検索エンジンの利用をスクリプト中に組み込むことが一つの解決策となる．(2) に関しては MetaCommander は従来のブラウザ操作に対するマクロ言語を提供しているとみなすことができるので，最もその効果が発揮される部分である．(3) に関しては MetaCommander では方針（B）で述べているように情報統合の単位を

(注3)：http://www.agentsoft.com
(注4)：http://www.metacrawler.com
(注5)：http://ahoy.cs.washington.edu:6060
(注6)：http://www.cs.washington.edu/homes/bobd/shopbot.html

ページよりも細かくしているので，一つのページに混在している情報を再構成し，機械的な情報解析が容易になるように構造化することが可能である．

2.2 システム構成

MetaCommanderの構成要素とデータの流れを図1に示す．利用者はMetaCommanderスクリプト（MC Script）を記述することにより，MetaCommanderに対して命令を与える．MetaCommanderはスクリプトを解釈し，その実行を行う．WWW情報源へのアクセスはインターネット（Internet）を介し，HTTP（Hyper Text Transfer Protocol）を用いて行われる．またローカルなデータ（Data）ファイルへの読み書きも可能になっている．MetaCommanderインタプリタにより得られた結果はHTMLテキスト（HTML Text）ファイルとして出力され，利用者はWWWブラウザ（WWW Browser）を用いることにより，その結果を表示させることができる．

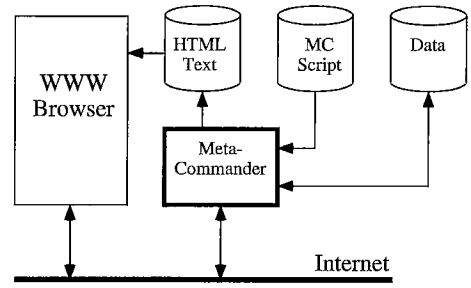


図1 MetaCommanderの構成
Fig.1 Components of MetaCommander.

2.3 MetaCommanderスクリプト

MetaCommanderで利用可能な主要関数の一覧を表1に示す．

表1 MetaCommanderの主要関数
Table 1 Major functions of MetaCommander.

分類	関数	機能
検索	getURL	URLからデータを得る
	postURL	CGIにポストしてデータを得る
	multipartURL	MIME形式でポストしてデータを得る
	fileURL	URLの内容をファイル化する
	password	ユーザ認証を指定する
抽出	getAnchor	アンカーを抽出する
	getString	タグを除いた文字列部分を抽出する
	searchString	文字列を含む文節を抽出する
	searchTag	タグを抽出する
	cutString	前後の文字列を指定して抽出する
整形表示	tag	タグを出力する
	print	標準出力へ出力する
保存	open	ファイルをオープンする
	eof	EOFに達したかを調べる
	getline	ファイルから1行読み込む
	putline	ファイルへ1行書き込む
	fprint	ファイルへ出力する
変数	set	変数を設定する
	unset	変数を削除する
	calc	変数に計算結果を代入する
	strcat	変数を連結させる
	chop	改行コードを取り除く
制御	if	条件分岐をする
	foreach	繰返し処理をする
	while	繰返し処理をする
	exit	強制終了する
	break	繰返し処理を抜ける
	continue	次の繰返し処理を始める

関数の書式はC言語に近いもので，関数名（引数1，引数2，...）となる．関数にはそれぞれの機能があるが，その効果は‘{’と‘}’で指定された範囲内でのみ有効であり，その範囲をスコープと呼んでいる．また，関数のスコープのあとに“else”と書き，‘{’と‘}’で囲むことで，その関数が正常に実行できなかったときに実行されるスコープを指定する．

例えば，

```
getURL( "http://www.ieice.or.jp/" ) {
    print
} else {
    print( "ERROR" )
}
```

は指定されたURLに正常にアクセスできた場合はその内容を表示し，そうでない場合は“ERROR”を表示する．

スクリプトに用意されている関数の中で最も特徴的なものは検索，抽出，整形・表示の関数群である．検索はURLにより指定されたWWWサーバからHTML文書を取り込むために用意されている．通常は引数をURLにしたgetURLを用いればよいが，フォームを用いてCGI（Common Gateway Interface）にデータを送る場合にはフォームの形式に応じてgetURL，postURL，multipartURLを使い分ける．ユーザ認証を必要とするようなサーバアクセスに関してはpassword関数を用いてユーザ名とパスワードの指定を行う．

抽出用の関数は取り込まれたHTML文書から必要な部分を切り取るために利用される．これにはリンクなどのアンカーを抽出するもの（getAnchor），タグを取り除くもの（getString），タグを指定して抽出するもの（searchTag），文字列を指定して抽出するもの（searchString，cutString）などがある．例えば，

```
cutString("ABC","XYZ") {
    print
```

}

は先頭が "ABC" で終端が "XYZ" であるような文字列を文書から切り出す命令である．

整形・表示用の関数としては print と tag がある．print は指定される文字列を標準出力（"meta.html" というファイル）に出力する．文字列の指定がなければスコープで指定される範囲の HTML 文書を出力する．tag は HTML 文書のタグを出力する関数で，出力される HTML 文書を整形する．

そのほか，通常の計算機言語に備わっているファイルアクセス，変数操作，数値/論理計算，繰返し操作などの関数と共にサブルーチン機能が用意されている．

3. ゲノムデータベースへの応用

ヒトゲノム計画（Human Genome Project）はヒトのもつすべての遺伝情報とその仕組みを明らかにしようとする国際的なプロジェクトである．このプロジェクトにより明らかにされる情報は遺伝病の解明や治療をはじめとして，生物学，医学，薬学の世界に大きな変革をもたらすものとして期待されている．一方でこのプロジェクトが抱える問題点は，膨大で多種多様な実験データをいかにして管理，利用するかという点であり，このためにデータベースをはじめとする情報処理技術はプロジェクトにとって不可欠の存在となっている[11]．現在では世界中に存在するゲノム研究者間での情報共有の手段として WWW が中核的な技術となっており，日本におけるゲノムネット(注7)のように，その上でさまざまなデータベースや解析ツールが公開されている．

このようなゲノムデータベースの特徴は大きく以下の三つにまとめられる．

a） 大量の情報処理

ゲノムデータベースがもつ第1の特徴はそのデータ量の多さである．最も低レベルの遺伝情報（ゲノム）は A, G, C, T という4種類の塩基の1次元配列として表すことができるが，ヒトの場合その長さは30億であると言われている．生物実験により大量に得られる断片的情報を手作業で収集，管理することは不可能であり，データベース化が必要不可欠となる．このような塩基配列のデータベースの一つである GenBank(注8) は 1997年5月16日（Release 100.0）の時点で，エントリ数にして 1,274,747，塩基数にして 842,864,309 の規模となっている．

b） データの更新

ゲノム配列の決定は日本，米国，欧州を中心とした世界中の研究所で行われており，データベースへの登録も日々行われている．先述した GenBank も日々更新が続けられており，差分を示す GenBank-upd は7月16日（Release 100.0+/07-16）の時点でエントリ数にして 251,602，塩基数にして 192,191,685 の規模となっている．すなわち，2か月の間にこれだけの量のデータが増加していることになる．そのために現在では WWW による情報提供の重要性が更に高まっている．

c） 多様な情報源

ヒトゲノム計画ではヒトのゲノム配列を決定するだけでなく，その構造や機能を解明することがその目的となっている．このためにはさまざまなデータベースや解析ツールが利用される．データベースには先に述べた GenBank のような DNA 塩基配列に関するものだけでなく，タンパク質や酵素に関するもの，遺伝病に関するもの，大腸菌，酵母，ショウジョウバエといった他の生物種に関するもの，文献に関するものなどがあり，研究者の目的に応じて使い分けられる．

また，あるゲノム配列の機能推定を行う場合にはすでに機能のわかっている類似配列がデータベース中に存在するかどうかを調べるホモロジ（相同性）解析が有効な手段であり，BLAST(注9) といった動的計画法に基づくアルゴリズムがデータベースと組み合わされて利用されている．

以上示したデータベースや解析ツール類はごく一部であるが，ゲノム研究者はこのような多様な情報資源を利用して研究を進めている．最近では特定の情報資源の単独利用だけでなく，それらを組み合わせて統合的に情報収集や解析を行うことの必要性も高くなってきている．

このようにゲノムデータベースは多くのものが WWW を介して利用可能となっているが，一方で WWW 技術の制約により情報の検索や解析のための多くの負荷をゲノム研究者に強いていることになっている．ここからは MetaCommander のゲノムデータベースへの応用と，その効果について述べる．

3.1 異種情報源の統合

先にも述べたように複数のゲノムデータベースを統

(注7)：http://www.genome.ad.jp
(注8)：http://www.ncbi.nlm.nih.gov/Web/Search/index.html
(注9)：http://www.genome.ad.jp/SIT/BLAST.html

合化する必要性が高まっており，このような試みの一つにゲノムネットにおけるDBget [1]がある．DBgetではゲノム解析に必要なさまざまなデータベースを統一的なインタフェースで利用可能にしている．またデータベース間の関連を表すLinkDBというリンク情報専門のデータベースを作ることで，リンクをたどることで相互に利用可能にしている．

しかしながらリンクによるデータベースの統合は操作性の点で問題がある．例えば核酸塩基配列データベースGenBankから文献情報データベースMedlineにはリンクが張られており，それをクリックすることで参照が可能であるが，参照したいエントリ数が増えるとその操作が煩わしくなる．これに対してMeta-CommanderではGenBankのデータ上のリンクを自動的にたどり，Medlineから必要な部分（例えば文献アブストラクト）だけを切り出してGenBankのデータ中に張り込ませることができる．このような統合は複数のデータベースをあたかも一つのものであるかのように見せることができ，リンクでの統合に比べてより利用しやすいものとなる．その具体的なスクリプトを図2に示す．

ここではまずGenBnakサーバのURL（1行目），検索エントリ数（2行目），キーワード（3行目）の変数への設定が行われる．続いてGenBankサーバに検索が行われる（4行目）．サーバはこの検索に対して図3(a)のようなエントリのリンクリストを返す．そこでそのリンクを切り出す（6行目）と共に，出力する（8行目）．このリンクの指すURLはシステム変数$_に代入されており，そのリンクをたどる（10行目）．サーバは図3(b)のような結果を返すので，そこから“REFERENCE”と“FEATURES”という文字列に囲

```
1:set( url, "http://www.genome.ad.jp/
  htbin/www_bfind_sub" )
2:set( max_hit, 5 )
3:set( keywords, "hiv human")

4:getURL($url,"dbkey"="genbank-today",
  "keywords"=$keywords, "mode"="bfind",
  "max_hit"=$max_hit) {
5:  file( "result.html" ) {
6:    getAnchor($max_hit) {
7:      tag("LI")
8:      print
9:      tag("BR")
10:      getURL($_) {
11:        cutString("REFERENCE","FEATURES",
           1,0){
12:          tag("PRE") { print }
13:          getAnchor() {
14:            getURL($_) {
15:              cutString("<P>","<P>") {
16:                if($COUNT == 3) { print
                   }}}}}}}}
```

図2　GenBankとMedlineの統合
Fig. 2　Integration of GenBank and Medline.

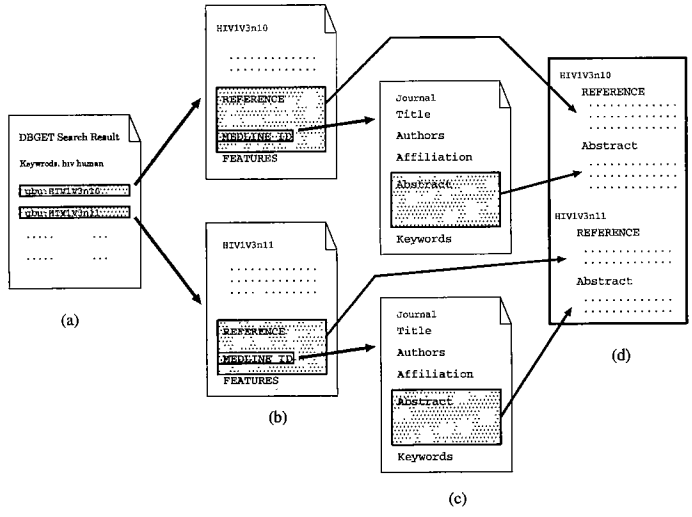


図3　GenBankとMedlineからの文献情報の収集
Fig. 3　Collecting reference information from GenBank and Medline.

まれた文献情報を切り出し（11行目），出力する（12行目）．続いてMedlineへのリンクを切り出し（13行目），Medlineサーバに検索を行う（14行目）．その結果（図3(c)）の第5パラグラフが文献アブストラクトであるのでそれを切り出す（15，16行目）．以上のプロセスを変数max_hitで指定された回数繰り返し（6行目），その出力をファイルresult.htmlに保存する（5行目）．その結果，図3(d)に示すようなHTML文書が得られる．

3.2 繰返し操作の自動化

BLAST等のホモロジ解析ツールはフォームを用いてパラメータや配列データを受け付けるようになっている．しかしながら解析したい配列データが多数あったり，パラメータの値を変化させて解析を行いたいときには，データファイルから何度もカットアンドペーストにより配列データをフォーム上にコピーしたり，パラメータボタンを変えてサブミットし直す必要がある．

このような何度も同じ操作を繰り返すような作業は図4に示すようなスクリプトとして記述することにより自動化することができる．すなわち解析すべき配列は1行ずつデータファイルseq.datに格納されており，それを変数sequenceに順次読み込む（4〜8行目）．変化させるデータベースパラメータdatalibの内容もforeach関数により順次切り換える（9行目）．そして配列データやパラメータが代わるごとに，データをサーバにサブミットすればよい（11行目のサブルーチンコール）．検索結果はデータサブミットごとに適当な識別子（ここではresult1，result2，...）を付けファイル化する（14行目）．

```
1:set( url, "http://www.ncbi.nlm.nih.gov:80/
 cgi-bin/BLAST/nph-blast" )
2:set( sequence , "" )
3:set( count , 0 )

4:open( "seq.dat" , fp , "r")
5:while( 1 ) {
6:  eof( fp ) { break }
7:  getline( fp )
8:  set( sequence , $_ )

9:  foreach(datalib , nr dbest dbsts) {
10:   calc(count, $count + 1 )
11:   searchBlast() }}

12:sub searchBlast {
13:  postURL( $url, "ADV_LAB"="  ",
     "PROGRAM"="blastn", "DATALIB"=$datalib,
     "FSET"="isset", "SEQUENCE"=$sequence,
     "PATH"="") {
14:   file( "result"$count ) { print }}}
```

図4 BLAST検索の自動化
Fig. 4 Automating BLAST search.

3.3 データベースの自動更新

ゲノム研究者にとって自分の興味をもつ領域のゲノム配列がデータベースに登録されているかどうか，またその機能はどこまで明らかになっているかは常に気にかかる事項である．そこで興味をもつ領域だけからなる私的なデータベースを作り，それを常に最新のものに更新したいという要求がある．

MetaCommanderではこのような要求も図5に示すようなスクリプトを記述することにより可能である．

```
1:set( url, "http://www.genome.ad.jp/
  htbin/www_bfind_sub" )
2:set( max_hit, 5 )
3:set( keywords, "human interleukin")
4:set( index, "index.txt" )

5:read_index()

6:getURL( $url,"dbkey"="genbank-upd",
  "keywords"=$keywords, "mode"="bfind",
  "max_hit"=$max_hit) {
7:  file( "result.html" ) {
8:    getAnchor( $max_hit ) {
9:      set( genurl , $_ )
10:     getString() { set( item , $_ ) }

11:     set( flag, "NO" )
12:     foreach( i , $LIST ) {
13:       if ( $i .eq $item ) {
14:         set( flag , "YES" )
15:         break }}
16:     if ( $flag .eq "NO" ) {
17:       add_index( $item )
18:       tag("LI")
19:       print
20:       tag("BR")
21:       getURL( $genurl ) {
22:         tag("PRE") { print }}}}}}

23:save_index()
```

図5 GenBankからの差分データの獲得
Fig. 5 Obtaining residual data from GenBank.

例えばGenBankでは先に述べたように，キーワード検索の結果は図3(a)に示すようなエントリの一覧として得られる．そこで初回の検索では，必要な情報をファイル化すると共に，エントリの一覧をインデックスファイルとして保存する．次回の検索からはまずインデックスファイルを読み込み（5行目のサブルーチンコール），得られたエントリの一覧とインデックスファイルに登録されたエントリを比較し（11〜16行目），登録されていないエントリに関してのみ結果を出力するようにする（18〜22行目）．また新たなエントリはインデックスファイルに追加し（17行目のサブルーチンコール），保存する（23行目のサブルーチンコール）．

4. 考　　察

4.1 既存の計算機言語との比較

他の計算機言語と比較した場合，MetaCommanderの特徴は以下の2点にまとめられる．

a） プラットホーム独立性

MetaCommanderで実現している機能の多くは従来のスクリプト言語によっても実現できる場合もある．例えば，WWWページのダウンロードを行うMetaCommanderスクリプト

```
fileURL("meta.html", "http://www.xyz.com/")
```

はMacintosh上ではWWWブラウザと以下に示すようにApple Scriptを利用することで可能である．

```
tell application "Netscape"
  activate
  GetURL "http://www.xyz.com/"
    to file "meta.html"
end tell
```

しかしこれには(i) Apple ScriptはMacintosh上でしか動作しない，(ii) WWWブラウザがApple Scriptに対応している必要がある，という制限がある．これに対してMetaCommanderはJava言語により実装しているために，プラットホームに依存せず，UNIX, Macintosh, Windows95などの代表的OS上で実行可能である．このプラットホーム独立性はインターネット上での協調を支援する上で重要な要因となる．今後，多くの利用者がMetaCommanderスクリプトを開発した場合には，それをプラットホームの違いを気にすることなく相互利用することができるという利益は大きい．但しプラットホーム独立性故の制限もある．例えば現在のMetaCommanderでは出力結果を直接ブラウザ上に表示することができず，それをいったんファイル化して手動で表示しなければならない．すなわちブラウザを操作はプラットホームに依存しており，この制限を除くためにはプラットホームに依存したコードを挿入する必要がある．

b） 簡　易　性

MetaCommanderの機能は当然のことながら既存のプログラミング言語でも記述可能である．これによりMetaCommanderにはできないような，より詳細な機能実現が可能になるであろう．一方でプログラム作成に要する作業量と必要な知識が増加する．例えば，図6は先述の`fileURL`コマンドをJava言語で記述した例である．ここではサーバとのコネクションを張るためのクラス（URL, URLConnection, DataInputStream）やメソッド（openConnection, getInputStream, readLine），ファイルアクセスを行うためのクラス（FileOutputStream, PrintStream），やメソッド（println），また例外処理のためのクラス（MalformedURLException, IOException）に関する知識がなければプログラムを行うことができない．

このようにMetaCommanderは利用者から計算機やネットワークの詳細な部分を隠ぺいしていることになる．これはゲノム情報処理のように，計算機の非専門家を利用者として想定する場合には特に重要であると言える．

4.2 エージェント技術による高度化

MetaCommanderを今後更に高度化させていく上でこれまでに研究されているエージェント技術を導入していくことが重要であると考えられる．Pattie Maes [8] はエージェントを特徴づける要因として個別性（personalized），自発性（proactive），適応性（adapted）を挙げている．そこで，このそれぞれの観点から考察を加えることにする．

a） 個　別　性

個別性とはエージェントが利用者の代理人としての働きをこなす機能である．ここではエージェントがいかにして利用者の意図を知るかという点が重要になる．これにはエージェント自らが学習する方法と利用者がスクリプトにより提示する方法の両極端があると考えられる．ゲノム情報処理の場合は利用者の要求は比較的系統だっており，そのあいまいさが少ないのでMetaCommanderではスクリプトによる方法を採用している．しかし，今後はいかにして少ない努力でスクリプトを記述するかという点が課題として残されてお

```
import java.io.*;
import java.net.*;

public class fileURL {
public static void fetchURL( String urlname ,
String filename )
throws MalformedURLException , IOException {

/* Open HTTP Connection */
URL url = new URL( urlname );
URLConnection conn = url.openConnection();
DataInputStream in = new DataInputStream(
conn.getInputStream() );

/* Open output file */
FileOutputStream fo = new FileOutputStream(
filename );
PrintStream out = new PrintStream( fo );

/* Download */
String line;
while ((line = in.readLine()) != null) {
out.println( line ); }}

public static void main( String args[] )
throws MalformedURLException , IOException {
fetchURL("http://www.xyz.com","meta.html"); }}
```

図6 fileURL の Java による実現
Fig. 6 An implementation of fileURL command by Java language.

り，これにはエージェントによる学習技術が期待される．WWW 上でのゲノム情報処理では，利用者の意図よりも情報提供を行う WWW ページの構造上でのあいまいさが大きい．例えば GUI を用いてスクリプトの自動生成を図るような場合は半構造的な WWW ページから利用者の意図としている部分をいかに切り出すかという点が重要な課題となる．これにはエージェントと利用者が相互作用しながら，ページの記述と利用者の意図を関係づける学習機構が有効な手段になると期待される．

b) 自 発 性

自発性とは利用者が知らない情報をエージェント側から提示する機能である．現在の MetaCommander はスクリプトに書かれたことを忠実に実行するだけであり，新たな情報を提示する機能はない．しかし例えば，指定された WWW サーバを巡回し，その変化を通知するようなスクリプトは容易に記述できるが，これは低いレベルで自発性を示す例であると言える．更に既存の検索エンジンと組み合わせれば，更に広い範囲の情報を収集し，より高いレベルでの自発性を実現することが期待できる．ゲノム情報学の分野では新しく得られた知見をできるだけ早く入手し，それを自らの研究に生かす努力が研究者に要求されており，質の高い情報を自発的に提供するエージェントへの期待は高い．

c) 適 応 性

適応性は外部の環境の変化に応じて適応的に動作する機能である．今後，MetaCommander のような WWW アクセスの自動化ツールが多く利用されるようになるとネットワークのトラヒックは更に増加することが予想される．現在のネットワーク利用形態は使いたい人が使いたいだけ利用できるという極めて利己的なものとなっている．これはネットワークだけでなく，サーバ利用に関しても同様である．前述の BLAST 等のホモロジ検索ではスーパコンピュータが必要になるほど多量の計算が必要となるが，多くの利用者が集中するために，高負荷の場合には電子メールによるバッチ処理を薦めている．一方で同様の BLAST サーバが世界中の多くの機関で運用されているが，それらの間で負荷分散を行うような仕組みにはなっておらず，世界的な視点からはこれらのサーバが有効利用されているとは言えない．このような問題に対してエージェントの適応的な機能は有効であろう．例えば，ネットワークやサーバの混雑を監視し，すいている時間帯を見計らってアクセスを行ったり，サーバと協調的に動作して，処理のスケジューリングを行うことも可能である．このように限られたネットワーク上の資源を有効利用するという立場から，エージェントの利己的ではなく，社会的な振舞いが今後求められることになるであろう．

5. むすび

スクリプトに基づき WWW 上での情報統合を行う MetaCommander について述べ，ゲノムデータベースへの応用を通してその有効性を示した．またプログラミングの観点から既存のスクリプトやプログラミング言語との比較を行った．更に MetaCommander を

更に高度化する上でのエージェント技術の可能性に関する考察を行った．ゲノム情報学の分野はWWWをはじめとするインターネット技術を積極的に利用した研究者ネットワークを構築しており，ネットワーク型情報処理の先駆け的存在であると言える．今後も更にネットワークを介した情報収集や統合に対する要求は更に増加すると共に，その高度化が予想され，MetaCommanderのようなシステムの必要性は大きくなるであろう．

なおMetaCommanderはWWW(注10)を介して公開中である．

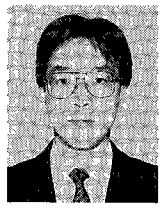
謝辞 本研究に対し貴重なコメントを頂いた京都大学大学院工学研究科の石田亨教授，三浦輝久君，中西英之君，京都大学化学研究所の金久實教授，国立がんセンター放射線研究部の谷上信室長，慶応大学医学部の蓑島伸生講師，東京大学ヒトゲノム解析センターの村上勝彦氏に感謝の意を表する．なお本研究の一部は文部省科学研究補助金重点領域研究「ゲノムサイエンス」によるものである．

文　献

[1] 秋山　泰，“WWWによる研究支援，”システム/制御/情報，vol.40, no.7, pp.291–296, July 1996.
[2] R.B. Doorenbos, O. Etzioni, and D.S. Weld, “A scalable comparison-shopping agent for the world-wide web,” Proc. 1st International Conference on Autonomous Agents, 1997.
[3] O. Etzioni, “Moving up the information food chain: Deploying softbots on the world wide web,” Proc. 13th National Conference on Artificial Intelligence, pp.1322–1326, 1996.
[4] H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom, “Integrating and accessing heterogeneous information sources in TSIMMIS,” Proc. AAAI Symposium on Information Gathering, pp.61–64, 1995.
[5] 金久　實編，“ヒューマンゲノム計画，”共立出版，東京，1997.
[6] B. Krulwich, “Automating the Internet: Agents as user surrogates,” IEEE Internet Computing, vol.1, no.4, pp.34–38, 1997.
[7] ローラ・リメイ，“HTML入門—WWWページの作成と公開，”プレンティスホール出版，東京，1995.
[8] P. Maes, “Pattie maes on software agents: Humanizing the global computer,” IEEE Internet Computing, vol.1, no.4, pp.10–19, 1997.
[9] E. Selberg and O. Etzioni, “Multi-service search and comparison using the metacrawler,” Proc. 4th World Wide Web Conference, pp.195–208, 1995.
[10] J. Shakes, M. Langheinrich, and O. Etzioni, “Dynamic Reference Sifting: A Case Study in the Homepage Domain,” Proc. 6th World Wide Web Conference, pp.189–200, 1996.
[11] 高木利久，“演繹データベースのゲノム情報処理への応用，”人工知能誌，vol.10, no.1, pp.17–23, Jan. 1995.
[12] G. Wiederhold, “Intelligent Integration of Information,” Kluwer Academic Publishers, Boston, 1996.

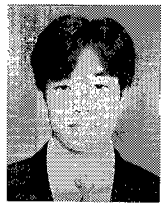
（平成9年7月29日受付，11月12日再受付）

(注10)：http://www.kdel.info.eng.osaka-cu.ac.jp/MetaCommander



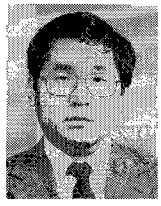
北村　泰彦　（正員）

昭58阪大・基礎工・情報卒．昭63同大大学院博士課程了．同年阪市大・工・電気助手．平2同大・工・情報講師．平8助教授．工博．平3英国キール大学客員講師．分散協調問題解決，ヒューリスティック探索，インターネットコンピューティングの研究に従事．情報処理学会，人工知能学会，ソフトウェア科学会，システム制御情報学会，IEEE，AAAI，ACM各会員．



野崎　哲也

平8阪市大・工・情報卒．平10同大大学院修士課程了．同年NTT入社．インターネットコンピューティングの研究に従事．



辰巳　昭治　（正員）

昭45阪大・工・通信卒．昭47同大大学院修士課程了．同年川崎重工業（株）入社．昭53同大学院博士課程了．豊橋技科大・助教授を経て，平2阪市大・工・情報助教授，平7教授．工博．統計的パターン認識，意思決定問題，画像処理用並列プロセッサの開発，VLSI向き相互結合網の構成法などの研究に従事．情報処理学会，人工知能学会，ソフトウェア科学会，IEEE等各会員．