地球流体物理科学者のためのデータアーカイブサーバ構築支援ツール：Gfdnaviにおけるデータベース設計と検索インタフェースの実装

柳平　有美†　　渡辺知恵美†　　堀之内　武††

† お茶の水女子大学理学部情報科学科　〒 112-8610 東京都文京区大塚 2-1-1
†† 京都大学生存圏研究所　〒 611-0011 京都府宇治市五ヶ庄
E-mail: †yumi@db.is.ocha.ac.jp, ††chiemi@is.ocha.ac.jp, †††horinout@rish.kyoto-u.ac.jp

あらまし　近年，地球流体物理科学データは爆発的に増加しており，科学者たちは自らが保有するデータから必要なデータを検索したり，科学者同士で互いに公開し合いたいという要求が高まっている．このような要求に応えるため，我々は科学者にかかるコストをできる限り削減することを目的とした地球流体物理科学者のためのデータアーカイブサーバ構築支援ツール：Gfdnaviの開発を進めている．本稿では科学データにおけるメタデータのスキーマ定義と，メタデータの自動生成，検索インタフェースの実装について述べ，さらには検索結果のグループ化とランキング手法について提案する．

キーワード　科学 DB，時空間 DB，データセンタ，ユーザインタフェース

Metadata Schema Design and Query Interface for Gfdnavi: A Data Archiving Server Construction Support Tool for Geophysical Fluid Database

Yumi YANAGITAIRA†, Chiemi WATANABE†, and Takeshi HORINOUCHI††

† Department of Information Sciences, Faculty of Science, Ochanomizu University
†† Research Institute for Sustainable Humanosphere, Kyoto University
E-mail: †yumi@db.is.ocha.ac.jp, ††chiemi@is.ocha.ac.jp, †††horinout@rish.kyoto-u.ac.jp

Abstract　In recent years, the earth fluid physical science data increases explosively and the needs has rizen that they needs to search required data appropriately from large amount of scientific datasets in their PCs, and share these datasets among them. To satisfy such demands, we develops Gfdnavi: a data archiving server construction support tool for geophysical fluid database. This system can cut the cost for scientists to construct data archiving server which services high functionalities for metadata search, analysis and visualization. In this paper, we describe about a metadata schema definition for scientific datasets, a automatic metadata extraction module, and a the query interface by using Google map. Particularly, in the query interface, we introduce some methods of grouping and ranking result data, the interface design can lead users to narrow search conditions to find their demanding data interactively.

Key words　Scientific DB，Spatial-temporal DB，Data Center，User Interface

1. ま え が き

近年，地球観測の測定機器の高機能化と計算機の高性能化により, 地球流体物理科学データは爆発的に増加しており，科学者たちは自らが保有するデータから必要なデータを検索したり，科学者同士で互いに公開し合いたいという要求が高まってきている．一般に大きな企業や団体，例えば NASA [9] や NOAA [13] ではデータセンタを設置して数百 TB 若しくは数 PB の膨大なデータを管理し，Web 上でデータを公開している．しかし，一般の科学者が自身でデータを検索・公開し合うためには作業コストや学習コストがかかる．そこで，低コストで尚且つ簡単に検索・公開を実現できるツールが必要とされている．このような要求に応えるため，我々は京都大学の堀之内武博士を中心とした地球流体分野のライブラリ開発チームであ

る地球流体電脳倶楽部 [3] と共同で，地球流体物理科学者のためのデータアーカイブサーバ構築支援ツール：Gfdnavi の開発を進めている．

Gfdnavi は大きくデータ検索部とデータ分析・可視化部，データ公開部の 3 つに分けられる．本稿では Gfdnavi の構成とデータ検索部について述べる．

まず第 2 節で Gfdnavi の特徴と構成について述べ，第 3 節ではデータ検索部のうち，科学データにおけるメタデータのスキーマ定義とメタデータの自動生成法について述べる．また第 4 節では，検索インタフェースの構成と検索を促す様々なインタラクションを提案する．最後に第 5 節でまとめと今後の課題を述べる．

2. Gfdnavi

我々が開発している地球流体科学者のためのデータアーカイブサーバ構築支援ツール:Gfdnavi は，地球流体科学者が個人で持つ膨大な科学データをローカルで検索したり，共同研究者や同分野の科学者同士でデータを公開し合いたいという要求に対し，それを簡単に実現することを目的としたツールである．

科学データの公開は従来，ある特定の公開者が提供する大規模なサーバやデータベースに対し多数の人が集中的にアクセスしデータを取得するという形でデータを公開していた．それから発展し，Web の登場・普及によって誰でも簡単にデータを HTML などの形で公開し，データを不特定多数のユーザで共有することが可能になった．しかし，データベースを利用した検索や公開など高度な機能を備えた公開サーバを提供したい場合，公開者には高い技術が求められ，なかなか実現できないというのが現状である．

これに対し，Ruby on Rails [14] という Ruby 言語による Web アプリケーション開発フレームワークがある．Ruby on Rails はデータベースとのやり取りを行う ActiveRecord(図 1Ⓐ) と Web サーバとのやり取りを行う ActivePack(図 1Ⓑ)，そしてそれらを連携させる ActiveSupport(図 1Ⓒ) で構成され，これを利用することによりデータベースをバックエンドに持つ Web サーバを容易に構築することができる．さらに Ruby on Rails は Ajax や Web サービスなどもサポートするなど高機能であり，現在注目を集めている．

Gfdnavi はこの Ruby on Rails を拡張し，地球流体科学者を対象とした高機能なデータカイブサーバ構築を支援するパッケージである．この Gfdnavi の特徴は以下のとおりである．

メタデータ自動抽出

公開ディレクトリのファイルから自動的にメタデータを抽出しデータベースに格納する．本システムは NetCDF や HDF などの地球科学データの標準フォーマットに対応しており，ユーザは対象フォーマットに保存したファイルを公開ディレクトリに保存するだけでよい．

高機能な検索・分析・可視化インタフェース

データの分析・可視化パッケージを豊富に提供するため，検索したデータに対して，様々な分析・可視化までをサポートする高機能な公開サーバが実現できる．

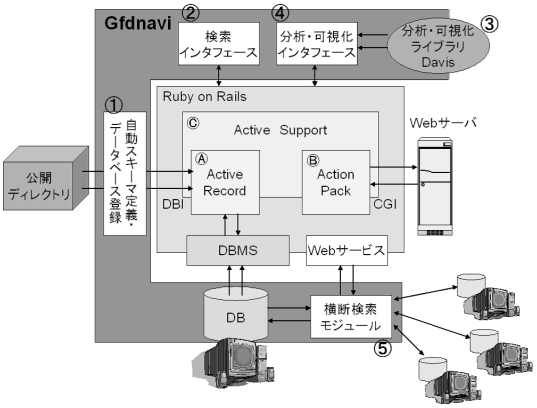

図 1　Gfdnavi の構成

デスクトップアプリケーションとしての個人利用

本システムはアーカイブサーバとして使うだけでなく，デスクトップサーチとして個人利用することもできる．地球流体分野の可視化ツールや分析ライブラリはあるものの，デスクトップ上のデータ管理から可視化，分析まで一連の処理を通して扱うシステムは貴重であり，デスクトップアプリケーションとしても有用である．

他の Gfdnavi との連携

Web サーバとして公開する他，他の Gfdnavi サーバとの横断的検索等の機能をもち，それぞれのサーバ間を横断的に検索してデータを共有することが出来る．

Gfdnavi は以下の 3 構成に大別できる．

(1)　データ検索部

公開用ディレクトリをスキャンしてデータファイルを自動的に認識し，メタデータを抽出して登録する (図 1①)．さらに，GoogleMap を用いた高機能な検索インタフェースを提供し，ユーザが手間なく検索を行えるようにする (図 1②)．

(2)　データ分析・可視化部

共同研究者の堀ノ内博士らは，これまで観測データをもとに分析・可視化するための Fortran ライブラリ DCL，Ruby ライブラリ Davis を提供している [4](図 1③)．これを Gfdnavi に搭載することにより，豊富なデータの分析・可視化を行えるようにする (図 1④)．

(3)　データ公開部

P2P を利用して個々に立ち上げている Gfdnavi サーバを横断的に検索し，個々が持つデータを容易に共有することを可能にする (図 1⑤)．

本稿ではこのうち (1) のデータ検索部について述べる．Gfdnavi 全体の概要および (2) のデータ分析・可視化部については [7] を，(3) のデータ公開部については [8] を参照していただきたい．

3. データ検索部の開発

データ検索部では，科学データのメタデータベース化および検索インタフェースの開発にかかるコストを出来る限り削減することを目的として開発を進めている．

もし科学者個人でデータを検索・公開したいという場合，デー

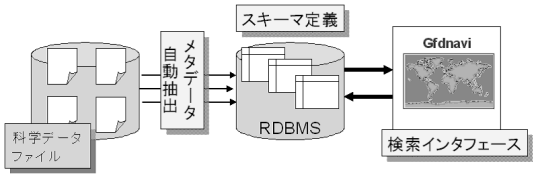

図 2　データ検索部

タのメタデータを RDBMS に格納するためのリレーショナルスキーマを定義し，科学データファイルに対するメタデータを定義して RDBMS に登録しなければならない．さらには，データを検索するためのインタフェースも作成する必要がある．これらのことを全て一般の科学者が行うには，作業コストや学習コストが非常にかかり実現するのは難しい．

この課題点に対し，まず我々はデータ検索部のイメージとして Google や Yahoo!のデスクトップサーチ [1], [2] のようなものを開発したいと考えた．デスクトップサーチは PC のアイドル時間にローカルコンピュータ内のファイルのメタデータを自動生成し，インターネットにおけるものと同じように検索したいキーワードを入力すればファイルを検索できる．ユーザは PC を起動させているだけで他に何もする必要が無い．

我々は科学データに対応するデスクトップサーチを想定し，次に示す方法によりデータ検索部の開発を行っている．

(1)　スキーマの定義

地球流体分野において広く使われるデータセットの構造に基づき，この分野において汎用的なスキーマを定義する．

(2)　メタデータの自動抽出

ユーザの指定した公開ディレクトリ下にある科学データファイルをスキャンし，ファイルヘッダ等の情報から自動的にメタデータを抽出してデータベースに登録するツールを開発する．

(3)　検索インタフェース

空間情報・時間情報・キーワードを検索パラメタとし，GoogleMap を用いたインタフェースを作成する．

これらのツールによるデータ検索部の流れを図 2 に示す．

現時点では，汎用的なメタデータのスキーマを定義し，それに従って科学データファイルからメタデータを自動的に抽出できる状態となっている．また検索においては，GoogleMap を用いたシンプルなインタフェースからの問合せに対して必要なデータを検索することが可能となっている．

以下 3.1 節でスキーマの定義について，3.2 節でメタデータの自動生成について，さらに 4 節で検索インタフェースについて詳しい内容をそれぞれ述べる．

3.1　地球流体データのためのメタデータ定義

前節でも述べたように，科学データには衛星データやゾンデデータ，レーダデータ，シミュレーションデータなどがあり，1 つのデータセットに関連するメタデータ情報としては計測範囲（緯度・経度）や計測条件，計算モデル，計算パラメタなど多様な属性が考えられる．地球流体データのデータフォーマットは従来は標準がなく非常に多岐にわたっていたが，近年標準化の試みが進められており，現在 NetCDF [10] や HDF-EOS [11] がその標準となりつつある．我々はこの分野において現在主

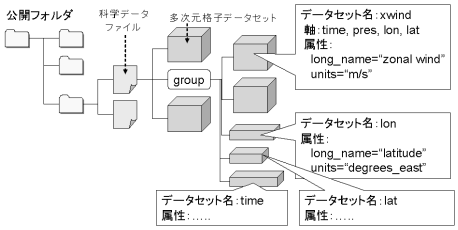

図 3　地球科学データフォーマットのデータ構造

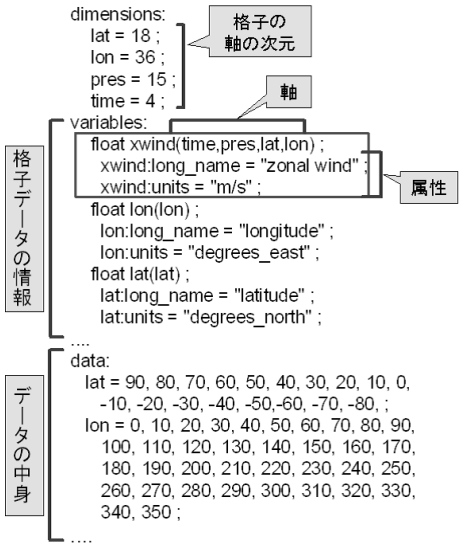

図 4　NetCDF ファイルのヘッダ情報

流となっているこの 2 種類のファイルフォーマットを元にメタデータ定義を行った．

NetCDF, HDF-EOS はどちらもメタデータを内包した自己記述型データフォーマットであり, 図 3 に示すように一つのファイルの中に複数のデータセットが構成されている．NetCDF は 1 階層であるが,HDF-EOS はグループオブジェクトを持ち階層的にデータセットを管理することができる．図 3 の例では，ファイルの中に xwind という風速データ, 気温データ, 緯度データ, 経度データなどが含まれている．

各々のデータセットには複数の属性を付与することができる．属性にはデータ観測における条件やシミュレーションにおけるパラメタセットなどが記述されている．データセットは多次元配列であり，n 個の 1 次元配列を互いに直行する軸として定義することにより n 次元格子を定義することが出来る．図 3 中の xwind データセットは long_name = ”zonal wind” units = ”m/s” を属性として持つ．図 4 は NetCDF ファイル形式のヘッダ情報である．このヘッダ情報には”dimensions:”以下に格子の軸となる配列の配列名が示されており，variables:以下に多次元格子データの情報が示されており，このファイルには xwind, lon, lat, pres, time という多次元格子データ (variable) がファイル内にあることが分かる。また

$$floatxwind(time, pres, lat, lon)$$

という記述により多次元格子データである xwind が time（時刻）,pres（気圧），lat（緯度），lon(経度）という一次元配列データを軸に持つ float 型の 4 次元格子をしていることが読み

取れる．また，
$xwind : long\_name =$ ”$zonalwind$”
$xwind : units =$ ”m/s”
は xwind に付与されている属性である．このようにして 1 つのファイルの中に複数の多次元格子を定義し，それぞれの格子データに属性を付与することが出来る．

また多次元格子データの格子構造は現在サポートしている NetCDF に限らず，基本的に以下の 3 種類に分けることができる．

- Grid 型：緯度・経度方向に平行/垂直な軸を持つ．
- Swath 型：衛星スキャン方向に平行/垂直な軸を持つ．
- Points 型：任意の点集合で表される．

この分類および定義は HDF-EOS によるものであるが，NetCDF における気象データのためのフォーマット規約である CF 規約 [12] においてもほぼ同様の格子に関する記述があり，またそのほかのデータ形式においても多少の違いこそあれ上記の格子構造のいずれかとして考えることができる．(ただし，HDF-EOS 以外のフォーマットでは上記の格子構造型が明記されていないため，データセットと空間軸の関係から格子構造を求める必要がある．これについては次節で述べる．)

これらのことを元に図 5 のような E-R ダイアグラムを設計し，以下のようにテーブルを定義した．

directories
(id int，parent_id int，name varchar，path varchar，plain_file tinyint)
variables
(id int,name varchar,directory_id int,path varchar)
spatial_attributes
(id int,variable_id int,directory_id,int,longitude_lb float,latitude_rb float,longitude_rt float,latitude_rt float)
time_attributes
(id int,variable_id int,start_time datetime,end_time datetime)
keyword_attributes
(id int,variable_id int,directory_id int,name varchar,value text)

本システムではデータセットのテーブルを variables とし，全ての科学データが持つ重要な属性として，空間属性 (spatial_attributes) と時間属性 (time_attributes) を抽出し，それら以外の属性値をキーワード属性 (keyword_attributes) として扱うこととする．また，それぞれのデータセットはフォルダごとにグループ分けされる場合もあるためディレクトリのテーブル (directories) を用意した．

以下空間属性テーブル (spatial_attributes)，時間属性テーブル (time_attributes)，キーワード属性テーブル（keyword_attributes）の定義について述べる．

【空間属性】
前述のとおり，格子構造は Grid, Swath, Points の 3 種類のタイプに分けることが出来る．これらの格子構造に対し，それぞれの空間属性は図 6 のように最小矩形の緯度・経度の最大値 (lat_rt,lon_rt) と最小値 (lat_lb,lon_lb) をとるように統一した．これにより格子構造に依存せずにデータセットの空間属性を表すことが出来る．

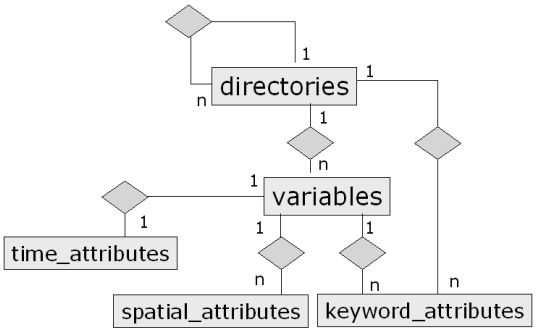

図 5　E-R ダイアグラム

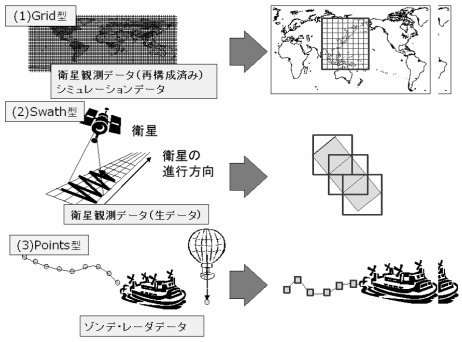

図 6　データ構造とその空間領域情報抽出

- Grid 型の場合（図 6(1)）：そのまま緯度・経度軸の最大値および最小値をとる．
- Swath 型の場合（図 6(2)）：長い帯状の格子を短い区間に区切り，各区間における緯度・経度の最大値と最小値をとる．この場合，1 つのデータセット (variable）に対して複数の空間属性が定義されることになる．
- Points 型の場合（図 6(3)）：緯度経度の最大値および最小値は同じ値となる．この場合も 1 つのデータセットに対して複数の空間属性が定義される．

【時間属性】
時間属性は 0 次元または 1 次元の配列であるため，その最小値を start_time，最大値を end_time とする．

【キーワード属性】
キーワード属性はデータセットそのものの説明から，測定パラメタ，シミュレーションパラメタなど実に多種類の属性が存在する．これらに対してはじめは代表的な属性を variable の属性として定義し，ファイルから抽出することも考えたが，どのような属性が定義されるかについてはファイルに保存されているデータの種類やファイル生成者の方針によっても異なる．そこでメタデータとして保存する場合には属性値を特定せず，属性名と属性値のペアという最も単純な形で保存させることとした．

以上の方針に基づき，図 3 の xwind からは以下のメタデータが抽出されデータベースに定義される．

variable(id,name,directory_id,path)
=(v001,xwind,NULL,”/filepath/filename:xwind/”)

spatial_attributes(id,variable_id,directory_id,
longitude_lb,latitude_rb,longitude_rt,latitude_rt)
=(s192,v001,NULL,min(longitude),
min(latitude),max(longitude),max(latitude))

time_attributes(id,variable_id,start_time,end_time)
=(t123,v001,min(time),max(time)

keyword_attributes(id,variable_id,directory_id,name,value)
=(k023,v001,NULL,"long_name","zonal_wind"),
(k024,v001,NULL,"units","m/s")

3.2 メタデータの自動生成

Davis ライブラリ [4] では，図 3 に示した NetCDF のデータ構造に基づきデータセットを分析ツールや可視化ツールで扱うための内部的なデータモデルとして GPhys モデル [6] を定義している．また Gphys モデルへの I/O クラスも数多く提供しており，NetCDF のほか GrADS, grib データなどに対応できる．

メタデータの自動生成モジュールは，現在 Davis ライブラリが対象とする上記のファイルフォーマットを対象とする．(現時点では HDF-EOS は対象から外れているものの，今後対象とする予定である.）本モジュールは公開ディレクトリにある対象フォーマットのファイルを Davis ライブラリで読み込み，そこから前節に述べた方針に基づいてメタデータを抽出することによって，Davis ライブラリが対象とするファイルフォーマットからメタデータが生成できる．但し，前節にも述べたように，HDF-EOS 以外のフォーマットでは格子構造型（Grid 型, Swath 型，Points 型）が明記されていない．Gphys モデルも同様に格子構造型は明記されていないため，データセットと空間軸の関係から以下のように格子構造型を求めた．

- **Grid 型**：データセットの格子軸の中に緯度軸と経度軸が含まれている場合は Grid 型である．
- **Swath 型**：Swath 型は衛星の進行方向に垂直な軸と平行な軸で格子が組まれるため，緯度経度は格子軸にはならない．この場合，緯度・経度データは衛星の進行方向に垂直な軸と平行な軸からなる 2 次元配列データとなり，緯度・経度データと同じ軸を含む多次元配列データセットの格子構造型は Swath 型となる．
- **Points 型**：1 次元配列の緯度・経度データの配列数が同じで，かつそれらと同じ配列数を持つデータセットは Points 型である．

メタデータの自動生成時間については，基本的にヘッダ部分のみをスキャンするためそれほど時間がかからない．しかしながら格子構造が Swath である variables に関しては空間属性を求めるために多次元配列そのものをスキャンしなければならないため今後高速化に関する工夫が必要であると考えている．

4. 検索インタフェース

検索インタフェースは図 7 のように，空間領域，時間領域，

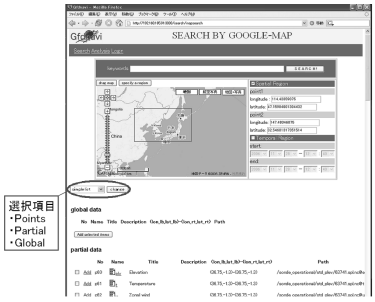

図 7　検索インタフェース

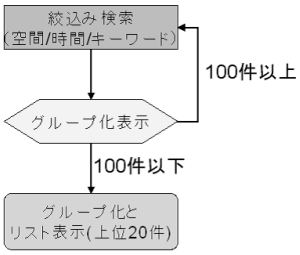

図 8　検索結果表示のフローチャート

キーワードのどれかを指定すると画面の下部に検索結果のリストが表示される．また，空間領域に対しては GoogleMap を用いて検索したい領域をドラッグするだけでも指定できるようにした．

しかし該当データが数百件もあった場合にそれを全てリスト表示するのは効率的ではない．そこで図 8 のフローチャートに示すように，検索結果が 100 件よりも多い場合はリストを表示せず，検索を繰り返してデータを 100 件以下に絞り込んだところでランキングを行い, 上位 20 件ずつリストを表示させるようにする．また，さらに簡単に絞込み検索ができるように，空間属性，時間属性，キーワードのそれぞれにおいて検索の絞込みを促すインタラクションを設計する．

以下 4.1 から 4.3 節で，空間属性，時間属性，キーワードのそれぞれにおける検索の絞込みを促すインタラクションの構想について述べ，さらに 4.4 節では，絞り込まれたデータのランキングについて述べる．

4.1 空間属性に対する検索

前節で述べたようにデータは格子構造ごとに Points 型, Swath 型，Grid 型に分けられるが，検索においてはこれらをある 1 地点の点データと，ある程度の範囲を持つ部分データ，そして地球全体をカバーする全球データに分類する．点データや部分データについては空間，時間，キーワードのどれにおいても検索することができるが，全球データは空間属性では絞込むことはできない．そこで，空間属性に対する検索においては，点データと部分データについてのみ考えることとする，

検索結果の表示については図 7 のように，GoogleMap の下に設けた選択切り替えを使用することで全球データ，部分データ，点データを分けて表示できるようにする．また，問合せに対する該当データを全て GoogleMap 上で表示すると繁雑になるため，それらをグループ化して表示させる．そして図 9 のように，GoogleMap をズームするごとにさらに細かいグループに分割して表示するようにする．

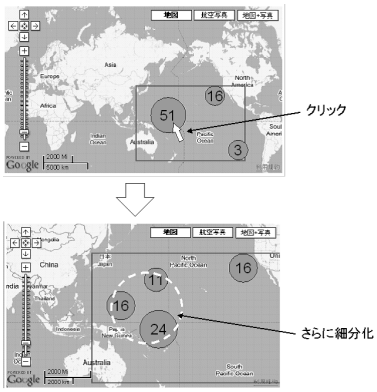

図 9　グループをズームして表示

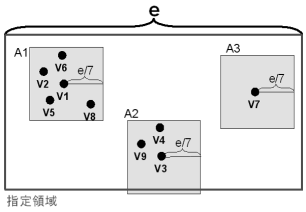

図 10　グループ化の例

さらに，該当データのリストと GoogleMap の表示をリンクさせ，GoogleMap 上で欲しいデータをクリックするとリストの該当するデータにチェックが入るようにし，逆にリストの方でクリックすると Map 上の該当する部分の色を変えるようにする．

4.1.1　グループ化のアルゴリズム

問合せ該当データは，最初に指定領域の中心に近い順にソートし，その後グループ化を行う．グループ化の表示は部分データと点データによって分け，アイコンの形も別々にする．

- 点データのグループ化（図 10）

ソートされた該当データ (v1，v2, …，vn) を順に見ていく．まず v1 をグループ A1 のメンバとし，v1 の緯度・経度をグループ化の基準値として使う．また，指定した問合せ領域の境界線のうち長い方の辺を e として，e/5 を閾値として扱う．この閾値は今後ユーザテストを行うことにより最も有効な値に変更したいと考えている．次に，v2 の緯度・経度をグループ A1 の基準となっているデータ v1 の緯度・経度と比べ，v2 が v1 の緯度（/経度）± e/5 の範囲内にある場合，v2 をグループ A1 のメンバとし，範囲外にある場合は v2 を新しいグループ A2 のメンバとする．v3, …，vn についても同様に，各グループ A1, A2 … Am の基準のデータの緯度（/経度）± e/5 の範囲と比べて判定し，グループに格納していく．最後に図 11 のように，GoogleMap 上でそれぞれのグループの基準データの緯度・経度を中心に円を表示し，各グループのメンバ数を表示する．さらに円の大きさはグループの件数に比例するようにする．

- 部分データのグループ化

部分データは点データの場合とほぼ同じアルゴリズムでグループ化を行うことを考えているが，同じグループであるかどうかの判定法として次の 2 通りのことを考えている．

(1)　部分データの中心点の距離を測り，点データの時と同様に閾値以内であれば同じグループとする．

(2)　部分データ v1 と v2 において，重なる領域がある場合は同じグループとする．

上記のどちらかの方法でデータをそれぞれの配列に分配し，点データと同様にそれぞれの配列を 1 つのグループとみなす．

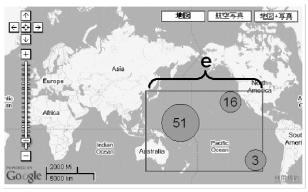

図 11　点データのグループ化表示

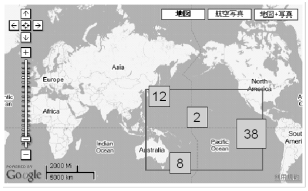

図 12　部分データのグループ化表示

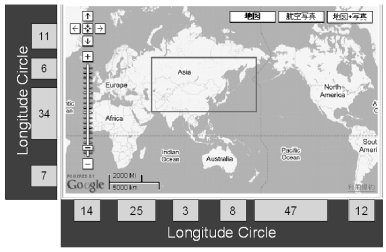

図 13　緯度円・経度円データの表示

4.1.2　緯度円・経度円のデータ

科学データの中には同一緯度 (経度) 円上のデータの平均をとったデータもある．このデータには緯度情報もしくは経度情報しか含まれていない．そこで，そのようなデータに対しては，図 13 のように点データと同じようなアルゴリズムでグループ化を行い，そのグループに含まれるデータの件数を GoogleMap の端に表示する．さらに，あるグループをクリックするとそのグループを細分化してデータのリストを表示するようにする．

4.1.3　複数の空間属性で構成されるデータ

複数のデータが一連となっているもの，例えば，swath 型や points 型のデータで，ある地点からある地点まで船舶で移動しながら一定時間ごとに観測したデータは同じ variable ファイルに含まれる．これらのデータに対しては次のような方法で Map 上に表示する．まず一連のデータが指定領域内から指定領域外につながって存在した場合，指定領域外にあるデータも問合せ該当データとする．また，グループ化したデータとは別扱いとして，同じ variable に含まれるデータのうちの 1 つがクリックされた場合は図 14 のようにそれと同じ variable のデータが全て数珠のようにつながるように表示し，それらが一連のデータであることが一目で分かるようにする．

4.2　時間属性に対する検索

時間属性に関しては全球データも検索の対象とし，空間属性に対する検索結果と同様にグループ化も行う．検索結果は図 15

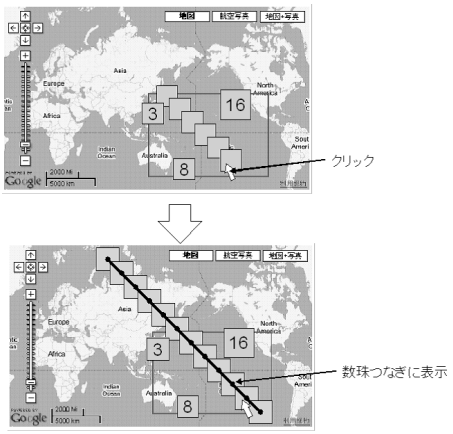

図 14　一連データのうち一つをクリックした場合の表示

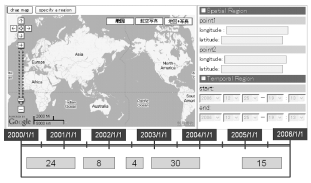

図 15　時間属性に対する検索の結果

図 16　キーワード入力のイメージ

のように時間軸を一定区間ごとに区切り，それぞれの区間に該当するデータ数を表示する．さらに，時間属性で絞込んだデータの中から空間属性についての検索ができるように，時間属性においても点データと部分データ，全球データは分類して考える．

4.2.1　キーワードに対する検索

キーワードに対する検索については図 16 のように選択項目を設けて，キーワードを簡単に入力できるようにする．

データの空間属性と時間属性以外の属性名は全てキーワードとすることとしたが，キーワードに含まれる属性名のうち上位 3 つを選択項目とし，さらにその属性名の属性値のうち上位 5 つを各選択項目の選択肢として設ける．これらの選択項目とその選択肢はデータを絞り込むごとに更新されるようにする．

4.3　ランキング

前節までで述べたようなインタラクションにより数十件に絞り込んだ検索結果のランキングを行い，適合率の高いものから順に表示する．ランキングには以下の 2 つの計算を取り入れることとした．

(1)　中心点の距離問合せ領域の中心点とデータの中心点の距離を計算する．これは Points 型のデータに対して特に有効になる．

(2)　問合せ領域に対するデータ領域の該当率問合せ領域に対してデータの領域がどれだけ重なっているかを計算する．これは Grid 型や Swath 型のデータに対して有効になる．

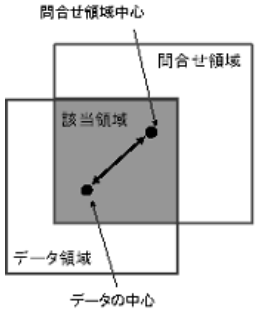

図 17　中心点の距離と領域の該当率

この 2 つの計算により全てのデータに対して有効なランキング計算を行うことができ，必要とするデータをより効率的に検索できると考える．

5.　まとめと今後の課題

本稿では地球流体物理科学者のためのデータアーカイブサーバ構築支援ツール：Gfdnavi のシステム構成のうち，データ検索部における汎用的なメタデータのスキーマ定義，メタデータの自動生成法，時間・空間情報を重要視した検索インタフェース，そして検索を促すインタラクションとして，検索結果のグループ化やランキング手法について提案した．現在本システムを開発中であるが，実際のデータへの適用や 4 節にて提案しているグルーピングやランキングに関する定量的な評価をするまでにはいたらなかった．今後は提案した検索インタフェースのインタラクションを実装し，提案手法の妥当性についての考察を行う予定である．またユーザテストを行うことによって 4.2 節で述べた部分データのグループ化の手法などについての検証を行うことも検討している．さらに，このツールをパッケージとして科学者に配布し，自由にユーザがカスタマイズできるようにしたいと考えている．これを実現できればデータ検索部の操作性と利便性をさらに向上させることが出来ると考える．

謝辞

本研究は, 文部科学省科研費特定領域「情報爆発時代に向けた新しい IT 基盤技術の研究」の課題 A01-14（課題番号 18049043）により行われた．

本研究遂行にあたって様々な協力やコメントを頂いた西澤誠也，森川晴大，林祥介，塩谷雅人氏ら地球流体電脳倶楽部の各氏に感謝する．

文　　献

[1]　Google Search Engine http://google.com
[2]　Yahoo! Desktop Search http://desktop.yahoo.com/
[3]　地球電脳倶楽部 http://www.gfd-dennou.org/
[4]　電脳 davis プロジェクト http://www.gfd-dennou.org/library/davis/index.html/
[5]　Chiemi Watanabe: “地球惑星科学研究者のためのデスクトップサーチツールの開発に向けて,” 情報処理学会研究報告 2006-DBS-140(2), Vol.2006, No.78, pp.429-436, 2006.
[6]　堀之内武，川那辺直樹 “Ruby による地球・惑星流体科学のためのプログラミング環境の開発”，情報処理学会研究報告 2002, Vol.43, No.1, pp.113-113, 2002.
[7]　堀之内武，西澤誠也，渡辺知恵美，森川晴大，神代剛，林祥介，塩谷雅人 “地球流体データベース・解析・可視化のための新しいサーバー兼デスクトップツール Gfdnavi の開発”，データ工学

ワークショップ (DEWS2007) , D2-8 (2007)
[8] 佐藤麻美，渡辺知恵美 “P2P を利用した地球流体データの横断検索・共有システムの実現に向けて”，データ工学ワークショップ (DEWS2007) , D1-9 (2007)
[9] NASA Goddard EOS Data Service http://eospso.gsfc.nasa.gov/eoshomepage/data_service.php/
[10] NetCDF(Network Common Data Form) http://www.unidata.ucar.edu/software/netcdf/
[11] HDF-EOS Project: http://hdf.ncsa.uiuc.edu/hdfeos.html
[12] NetCDF Climate and Forecast Metadata Convention http://cf-pcmdi.llnl.gov/
[13] NOAA Satellite and Information Service http://www.class.noaa.gov/nsaa/products/welcome/
[14] Ruby on Rails http://www.rubyonrails.com/