

対話的な絞込み操作を考慮した P2P による 地球流体データアーカイブサーバの横断検索

齋藤 真衣[†] 堀之内 武^{††} 渡辺知恵美^{†††}

[†] 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 宇治市五ヶ庄 京都大学生存圏研究所

^{†††} お茶の水女子大学大学院人間文化創成科学研究科 〒112-8610 東京都文京区大塚 2-1-1

E-mail: [†]maisaito@db.is.ocha.ac.jp, ^{††}horinout@rishi.kyoto-u.ac.jp, ^{†††}chiemi@is.ocha.ac.jp

あらまし 近年増加している科学データに対し、個人レベルでのストレージの確保やデータの分類をすることなくデータ検索・公開・解析・可視化をサポートし、リレーショナルデータベースを用いたアーカイブサーバを構築できる Gfdnavi を開発している。Gfdnavi が一般配布され多くのアーカイブサーバが Web 上に出現した時、個々のサーバでの検索だけでなく、P2P を用いて他の公開サーバとの横断的検索をすることで研究者同士の知識共有を可能にする。そこで我々は Gfdnavi の横断検索の開発を進めてきた。Gfdnavi で共通に定義されているリレーショナルスキーマをもとに、DHT を利用した絞込み検索を行うことで結果候補を所有するノードを求め、Gfdnavi の Web サービスを介して該当ノードに問合せを行うことによって複数のノードから検索結果を取得する。実用段階に向け Gfdnavi の探索的検索インタフェースからシームレスに横断検索ができるようビット配列を用いた絞込みを可能にして拡張を行い、またグループ内でのみの共有ができるようにアクセス制御についても考慮した。

キーワード 科学 DB, P2P, オーバーレイネットワーク

A cross-search mechanism for Gfdnavi

Mai SAITO[†], Takeshi HORINOUCI^{††}, and Chiemi WATANABE^{†††}

[†] Department of Information Sciences, Faculty of Science, Ocyanomizu University

^{††} Research Institute for Sustainable Humanosphere, Kyoto University

^{†††} Graduate School of Humanities and Sciences, Ocyanomizu University

E-mail: [†]maisaito@db.is.ocha.ac.jp, ^{††}horinout@rishi.kyoto-u.ac.jp, ^{†††}chiemi@is.ocha.ac.jp

Abstract In recent years, earth fluid physical science data increase explosively. Owing to the rapid technological development of computers, even individual scientists own large-scale scientific repositories in their labs or PCs, and there are growing demands for retrieving desired datasets from such repositories. Furthermore, from the viewpoint of e-Science, trading such scientific datasets among scientists in the same or related domains is crucial. To satisfy such demands, we have been developing Gfdnavi, a tool for supporting construction of data archiving server for geophysical fluid database. Gfdnavi provides utility functions, such as metadata extraction, highly-functional query interface, and data analysis/visualization libraries. Additionally, Gfdnavi has a cross-search mechanism by which a user can search across several Gfdnavi servers beyond the Net. In this paper, we describe the cross-search mechanism of Gfdnavi. The system aims at an autonomous, decentralized retrieval system using the P2P network without setting up a central server. Gfdnavi provides faceted navigation interface to search scientific data, and we propose a p2p search mechanism based on a user-computer interaction flow in faceted navigation interface.

Key words Scientific DB, P2P, Overlay Network

1. はじめに

近年の地球観測と計算機の進展により大気や水質などの数値

データは爆発的に増加している。これらの数値データは一般的に多次元・多量であり解析・可視化して初めて意味を持つ。NASA などはデータセンタを設置し場合によっては数 PB にも

及び地球観測データを管理し Web 上で公開しており、世界中の科学者がダウンロードし利用できるよなになっている。

このような背景の下で科学者個人が管理するデータは飛躍的に増加し、また研究者個人で観測したデータそのものも大きな容量を持っている。そこで効率の良い研究のために気軽に自分の計算機に蓄積されたデータを検索したり、同様の研究をする科学者に観測データを公開したりしたいという要求は高まっているが、個人レベルではストレージの確保や自主的なデータの分類作業など様々な手間が掛かり、二の足を踏む科学者も多い。この現状を改善するべく、個人やグループ内におけるデータ解析・可視化作業からデータ検索・大規模データ公開までをカバーする Gfdnavi を共同で開発している (図 1)。

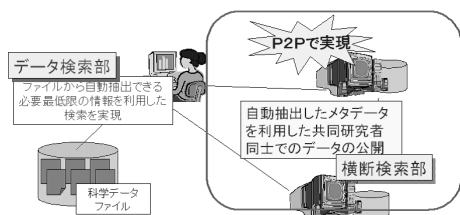


図 1 Gfdnavi のシステム構成

Gfdnavi はリレーショナルデータベース (RDB) を用いた地球流体データアーカイブサーバを比較的容易に作成することができる。Gfdnavi が一般配布されれば、多くの研究者によるアーカイブサーバが Web 上に出現することになるだろう。その時、個々のサーバでの検索だけでなく、P2P を用いて他の公開サーバとの横断的検索ができれば研究者同士の活発な知識共有が期待できる。そこで我々は Gfdnavi の横断検索の開発を進めてきた。先行研究 [3] では空間条件とキーワードによる横断検索を試験的に実装したが、実用段階に向け Gfdnavi の探索的検索インタフェースからシームレスに横断検索ができるように拡張を行った。またグループ内でのみの共有が可能となるよう、アクセス制御についても考察した。

本稿では Gfdnavi における横断検索について述べる。2 節で本研究の目的である P2P を利用した横断検索の前提知識となる分散ハッシュテーブル (DHT) と Gfdnavi で想定される利用状況、先行研究について、3 節で Gfdnavi における横断検索について、4 節で DHT のデータ構造と処理、5 節で実装、6 節でまとめと今後の課題を提示する。

2. 前提知識

本節では、Gfdnavi のピュア P2P 横断検索にて採用している分散ハッシュテーブルと Gfdnavi で想定される利用状況、さらに DHT を用いて Gfdnavi における横断検索を検討した先行研究について述べる。

2.1 DHT(Distributed Hash Table)

DHT はハッシュ関数を通して算出したキーとそれに対応する値の組を格納したハッシュテーブルを P2P ネットワークに参加したノードで分散して管理し、その分散されたハッシュテーブルを辿って効率的に検索を行う手法である。ノードが P2P ネットワークに参加すると一意な ID が付与され、ノードが公

開したデータの情報 (メタデータ) がハッシュ関数を用いて生成される。メタデータは「データを特定するもの」をキーとし、「該当するデータもしくはそのデータの所在地 (ノード id, オブジェクト id)」を値とする 2 項の組である。各ノードは自分の ID と同じ、もしくは近隣のキー値のハッシュテーブルを管理する。またネットワーク上に存在する幾つかの他ノードの所在情報 (スキップリスト) を管理する。検索時は、検索したいデータのハッシュ値を算出し、所在を知っているノードのうち探しているハッシュ値に最も近い値の ID を持つノードへ探索依頼を送る。ID やキー値の付与の仕方やメタデータの管理、ノードの辿り方 (ルーティング) のアルゴリズムには Chord, Pastry などがある。DHT でユーザが使う関数は put と get である。put は (キー, 値) のペアを指定して DHT に登録し、get はキーを指定することでそれに対応する値を返す。

2.2 Gfdnavi で想定される状況

Gfdnavi は Web ブラウザインタフェースを用いたアプリケーションであり、スタンドアロン型の科学データ分析ツールとして利用できるが、同じインタフェースのまま Web コンテンツとして一般に公開することもできる。さらに横断検索を導入することにより、Web サーバとしては公開せずに P2P ネットワークを通じて仲間うちで利用することも想定される。図 2 に Gfdnavi で想定される状況を示す。

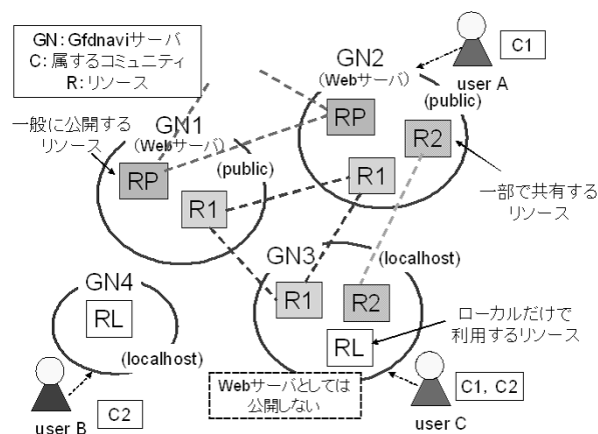


図 2 想定される状況

Gfdnavi サーバ GN1~4 のうち、GN1~2 は一般公開するリソースを所持する公開サーバ、GN3~4 はローカルで利用するリソースやグループ内で共有するリソースを所持するサーバである。コミュニティ C1 に属する userA は一般公開する RP の他に C1 内で共有する R1 にアクセスする。同様に userB は RP に加え GN4 でローカルに利用される RL、C2 で共有される R2、userC は RP、RL、R1、R2 にアクセスする。リソースを一般公開する場合、Web サーバを起動し DHT に登録することで全ての Gfdnavi から P2P ネットワークを通した横断検索が可能になり、単体の Gfdnavi サーバとして利用する場合は DHT に登録しなければ他から横断検索されることはない。また DHT に登録する際にアクセス制御を考慮することで、限られた一部の Gfdnavi におけるリソースの共有を実現する。

2.3 先行研究

先行研究 [3] にて佐藤らは、2 段階の処理を経て問合せを実行した。図 3 にそのイメージ図を示す。

まず想定する問合せを限定し、ユーザが指定する空間条件とキーワードをキー、その条件を満たすノードを値として DHT に登録して、それぞれの条件に該当するデータを所持するノードを DHT に問い合わせて特定する。また各 Gfdnavi サーバには SQL を受け付けその結果を返す Web サービスを起動しておき、DHT 検索で該当データを所持しているとされた全ノードに対し、Web サービスを用いて SQL を発行する。DHT で行われるのはデータを所持するノードを特定するだけの検索であり、ほとんどノードが絞られない場合は多くのサーバに SQL を発行しなければならない。さらに Web サービスでは任意の SQL を実行できてしまい、データの削除も可能となるためセキュリティ上に問題がある。そして対象ノードがネットワークに参加していない場合には問合せに対処できない。また先行研究では試験的に横断検索を導入したため Java によるコマンドプログラムを実行しており、Gfdnavi から動作させることまで具体的に想定していなかった。

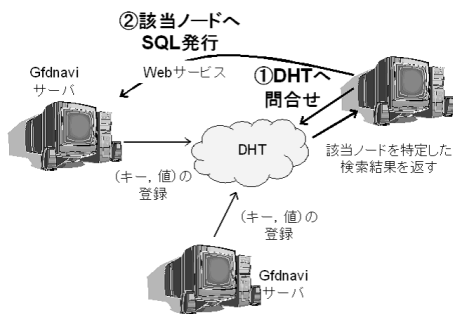


図 3 2 段階の問合せ処理

3. Gfdnavi における横断検索

Gfdnavi では図 4 のようなメタデータに対する属性が定義されており、ユーザは各属性リストから検索条件を指定することで検索を進める。各属性の詳細や属性間の関連については [6] を参

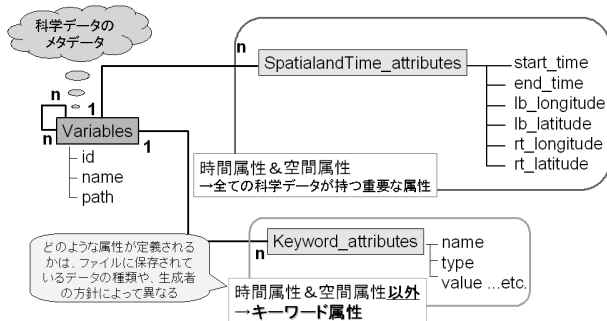


図 4 Gfdnavi における属性

照されたい。前節で述べた先行研究の問題点を踏まえ、Gfdnavi の横断検索機能の実用化に向け本研究では Gfdnavi に新しく加わる機能として重川 [6] により実装されている「探索的検索インタフェース (以下 Gfdexplorer と呼ぶ)」を、そのまま横断検索

でも利用できることを目的とする。図 5 は Gfdexplorer のスクリーンショットである。①はキーワード属性リスト、②は結果データセット情報、③は該当オブジェクトの位置情報、④は時間属性リストを示している。ここで探索的検索とはあるデータ全体の中から必要としているデータを求める際に 1 度検索をかけるだけでなく、条件によってデータを絞込んでいく検索を繰り返したり、検索結果を比較したり、また検索結果同士を組み合わせたりしながら徐々にユーザの必要とするデータに近づいていく検索方法を指す。既存の Gfdnavi では空間属性に対してのみ GoogleMap を用いた探索的検索ができるが、時間属性やキーワード属性に対しても同様に探索的検索が行えるよう現在新しいインタラクションモデルが開発されている。新しい検索インタフェースではデータの属性値をもとにグルーピングをし、その結果をリスト表示する。ユーザがリスト中のグループの一つをクリックするとそれを絞り込み条件として問合せをし、データを絞り込んで結果を出力する。このインタフェースで横断検索をする際、ユーザは図 5 の右上部分の *enable cross_search* にチェックを入れるだけで良いものとする。

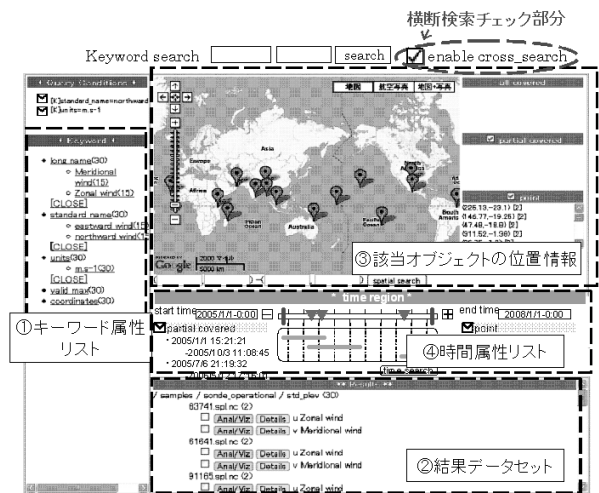


図 5 Gfdexplorer スクリーンショット

3.1 基本アイデア

本項では Gfdexplorer で横断検索をするための基本的アイデアについて述べる。2.1 節で述べたように DHT ではキーで検索し、対応する値を結果として返す。そこで DHT での検索のために考えるべきことは何をキーにし、何を値とするかに帰結する。図 6 に検索の流れを示す。Gfdexplorer では検索対象のデータセットの概要情報 (主要なキーワード属性名と各々のキーワード属性のとり得る主要な値、空間属性、時間属性) がリストアップされており、そのいずれかをクリックするとその値で絞り込み検索が実行され、結果データセットの概要情報が表示される。DHT における検索でも概要情報に表示されている全ての要素をキーとし、各キーについてそのキーをクリックした時に表示される

- ・キーワード属性リスト (図 5①)
 - ・結果データセット情報 (図 5②)
- を値とする。キーワード属性リストについては 4.1 節、結果データセット情報については 4.2 節で詳しく説明する。また、上記

のみでは複数の検索条件による絞り込みができないため、絞り込み操作のためのビット配列も値として DHT に登録する。ビット配列による絞り込みについての詳細は 4.1 節で述べる。

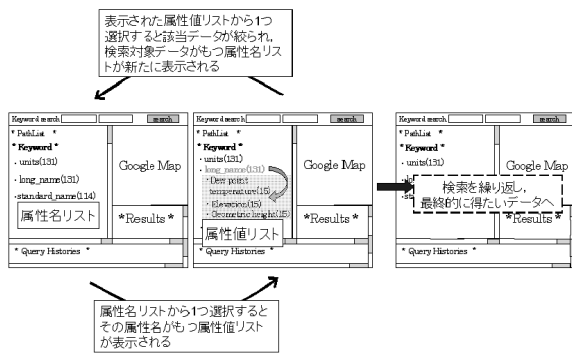


図 6 Gfdexplorer における検索の流れ

4. 探索的検索を実現するための DHT のデータ構造と処理

本節では探索的検索を横断検索において実現するため、各属性における DHT のデータ構造と処理について述べる。

4.1 キーワード属性リストの表示と絞り込み

図 7 に Gfdexplorer におけるキーワードによる絞り込み検索の流れを示す。Gfdexplorer ではユーザーがリストから条件を 1 つ

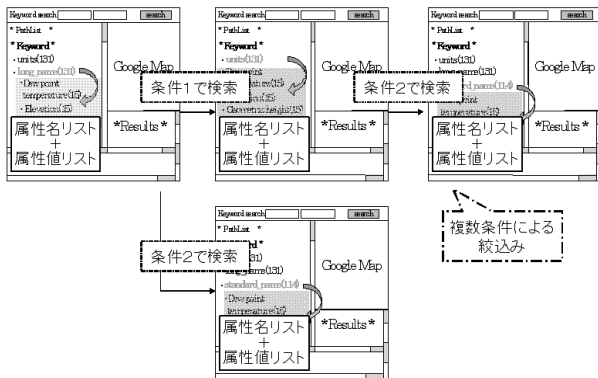


図 7 絞り込み検索の流れ

指定すると、それを満たすリストとその件数を結果として返す。例えばリストから属性名 standard_name を選択すると、その属性値 Dewpoint temperature や Elevation の他にそれぞれの該当件数が表示される。さらに属性値 Elevation を選択すると属性名 standard_name の属性値 Elevation で検索され、次の絞り込み条件の候補となる属性名リストが返される。3.1 節の基本アイデアに基づき、横断検索でも同様に検索結果リスト (図 5①) が返されるが、問題となるのは複数条件による絞り込みを行う場合である。もし図 7 において条件 1 による検索結果が 10 件、条件 2 による検索結果が 5 件であった場合、該当件数の少ない条件 2 による検索結果をとって見積もることはできるが、これは正しい絞り込みではない。そこで本研究では各 variable が検索条件を満たすか満たさないかを 1 と 0 だけのビット値で表したビット配列を用いる。例として図 8 に示されるキーワード属性

における絞り込みでは、属性名 A の属性値 a3 を満たす variable に対応するビットは 1 で表されている。データの有無を 1 ビットで表現でき、ビット間の論理積や論理和の計算は高速であることから、ビット配列を用いて絞り込み検索を実現する。まずキーワード属性における検索ではユーザーが検索条件として指定し得る属性名、さらにその属性値それぞれに対して variable 数の分だけビット配列を用意し、各ビットは各 variable_id に順に対応させる。例えば図 8 では属性名 A、属性名 A の属性値

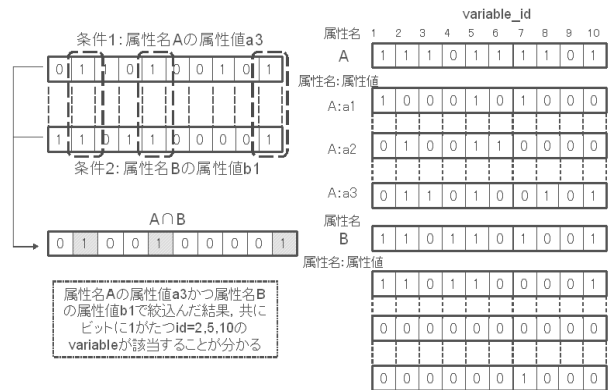


図 8 キーワード属性における絞り込み

a1, a2, a3 に対してビット配列を用意し、属性名 A の属性値 a3 かつ属性名 B の属性値 b1 で絞り込まれた場合、共にビットに 1 がたつ id=5 の variable が該当する。その結果をもとにキーワード属性のリストとその件数を求める必要がある。絞り込みの際の該当件数取得の流れを図 9 に示す。クライアント側で属性

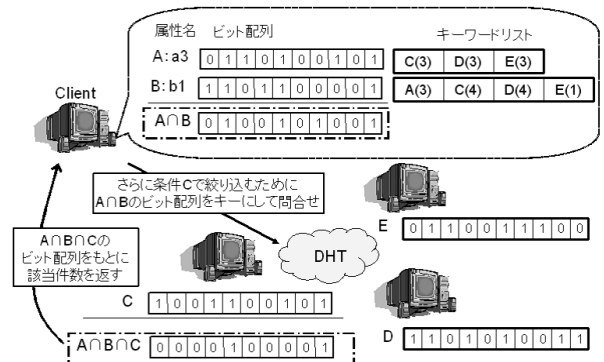


図 9 該当件数取得の流れ

名 A と B のビット配列、キーワードリストを所持し、そのリストから共通する属性 C, D, E について絞り込み可能であることから、属性 C で絞り込むこととする。キーワードリストとその件数から A かつ C は 3 件、B かつ C は 4 件であり、A かつ B かつ C は 3 件と見積もることはできるが、実件数は求めることができない。この場合 A かつ B のビット配列をキーとして DHT に問い合わせ、A かつ B かつ C のビット配列をもとに該当件数を計算する必要があるが、クライアント側でビット配列をすべて取得してくるのはコストが高い。そこで我々は、ビット配列を置いてあるノードで該当件数処理をする getcount 関数を用意する。件数取得のチェック項目を設け必要に応じて件

数取得処理をすることで DHT で取得するデータ量を減らす。getcount は以下のように定義する。

```
int getcount(key,bitmap)
```

検索条件とクライアントが所持するビット配列をキーとし、次の絞込み条件を満たすビット配列をもとに該当件数を値として返すものとする。図 2 の例では最初の見積もりとは異なり、該当件数 2 件と分かる。このように各キーに対して検索結果リスト、ビット配列を DHT の値とすることで、キーワード属性における絞込み検索ができる。ここでは説明のために単一ノードから結果を得る場合を例としたが、横断検索においては複数ノードから検索結果が返ってくることが考えられる。その場合、複数ノード分のビット配列を取得してクライアント側で所持しておき、ノードごとに getcount で件数を計算してその合計を該当件数として返す。Gfdnavi 全体では大きなノード数を扱うが、すべてのノードに対して横断検索するのではなく図 2 のように実際は共有ノード間と一般公開ノードによるものと想定されるため、そのビット配列を取得するコストはそれほど高くないと考えられる。また該当ノードをビット配列で表すときはデータの挿入、削除、更新に対してコストがかかるという問題がある。ただし Gfdnavi の横断検索では现阶段ではそれらの更新は頻繁に行われないものと仮定する。

4.2 オブジェクト

本項では結果データセット情報(図 5②)を表示させるために DHT に登録するデータ構造について述べる。まず variable_id をキー、オブジェクトの id を値として DHT に登録し、オブジェクトの基本情報を取得できるようにする。各属性について検索を進めるとビット配列から該当 variable_id が割り出されるので、ビットの位置情報をキーとし、基本情報、ノード id、名前、パス、description を値として登録する。ブラウザの表示範囲の大きさから最大 5 件ずつ表示するものとし、1 で表されるビットのうち 5 つを選んでオブジェクト情報を取得する。

4.3 空間属性

空間属性においては緯度経度をそれぞれ 10 度ずつに切り分け、各々ビット列を用意する。ユーザが図 10 のように検索したい空間を指定した場合、その空間の緯度経度の範囲を一部でも含めば条件を満たすものとし、経度は 10~30 度、緯度は 20~50 度の範囲のビット列を参照する。緯度経度ごとにビッ

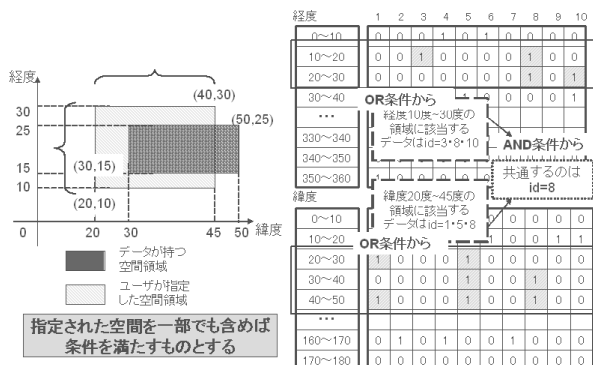


図 10 空間属性における絞込み

トの論理和をとると、経度は id=3,8,10 の variable、緯度は

id=1,5,8 の variable が該当する。最後に緯度経度共に条件を満たす variable を論理積で求めると id=8 の variable が該当し、これにより空間絞込みができる。また検索の初期段階では該当データ数が多く、ビット配列による絞込みのコストが高くなってしまったため、まず大まかに 60 度ずつに切り分けて絞込みが進んでから細かく切り分けることとする。

4.4 時間属性

時間属性においても年、月、日毎にビット列を用いる(図 11)。2008 年 2 月 1 日という時間属性で絞り込む場合、年月日毎に対応ビット配列を参照すると図 11 では id=2 の variable がすべてのビットで該当することが分かる。

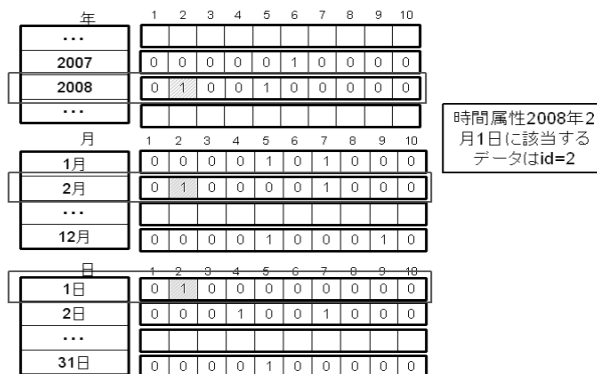


図 11 時間属性における絞込み

5. 実装

Gfdnavi は Ruby 言語による Web アプリケーション開発フレームワークである 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クサーバ構築を支援するパッケージである。横断検索部では P2P ネットワークルーティング及び分散ハッシュテーブルの管理・制御に Java 言語によって実装された OverlayWeaver [5] を利用するため、Ruby から Java のクラスを呼び出すライブラリである RJB(Ruby Java Bridge) [7] を用いてデータ検索部との連携を図る。我々は RDB に格納されているテーブルの索引情報を OverlayWeaver の分散ハッシュテーブル管理モジュールに登録するハッシュ検索登録モジュールと、検索インタフェースにより実行された問合せをもとに横断的検索を行う問合せ処理モジュールを実装する。

5.1 DHT への登録

図 12 に DHT への登録の流れを示す。まず 3.1 節で述べた基本アイデアに基づき、RDB に格納されている検索対象データセットからユーザが検索条件に指定する DHT のキーと、それに対応する値を生成する。ここで DHT に登録する値のうちブラウザ上で表示される検索結果リストはこの段階では複数の情報をまとめて格納している型であるため、これを DHT の値として put するためには文字列に変換する必要がある。図 12 の結果生成モジュールで文字列化した値を OverlayWeaver で put することで、キーと値の組が DHT へ登録される。

5.2 DHT を用いた検索

図 13 に DHT を用いた検索の流れを示す。横断検索時、まず step1 としてユーザが一覧から指定した属性名がインタフェー

スからパラメータとして取得され、横断検索モジュールに渡されて DHT のキーが生成される。そこで Ruby で実装されている横断検索モジュールから RJB を用いて OverlayWeaver を呼び出し、get 関数を実行してキーに対応する DHT の値を取得する。取得された値は結果解凍モジュールで検索結果として表示できるように変換される。次に step2 では step1 によって表示された結果に対してユーザが該当件数取得したい場合、チェックの有無によって該当件数を求める処理が行われる。このような 2 段階の問合せを繰り返すことにより、インタフェースにはユーザが指定した条件で絞り込んだ検索結果が表示される。

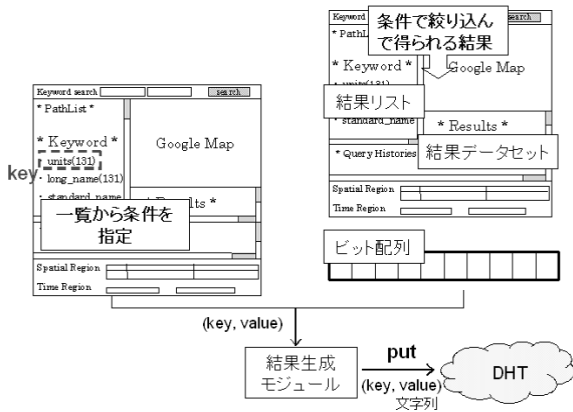


図 12 DHT への登録モジュール

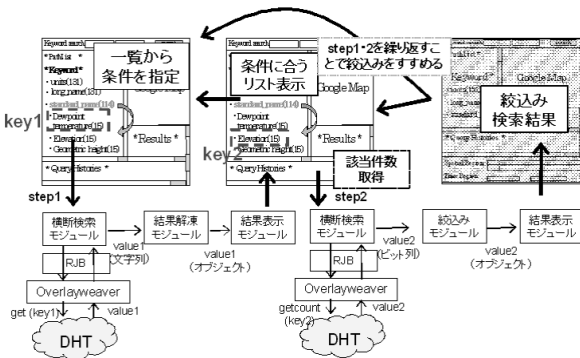


図 13 DHT への問合せ処理モジュール

5.3 暗号化によるアクセス制御

図 14 に暗号化による DHT のアクセス制御の流れを示す。2.2 節で述べたように、Gfdnavi で扱われるデータの中には例えば共同研究者間など一部のユーザのみに公開したいデータが存在する状況が考えられる。横断検索によって共有を望まないデータまでもが他人に公開されてしまうことを防ぐため、データを共有するためのグループを設定し、グループ単位のアクセス制御を実現する。まずグループごとに鍵を用意し、データの所在を示す DHT の値を暗号化する。横断検索時に必要な DHT のキーにグループ名も追加し、データを共有する権利があるかどうか判断する。グループに属していれば、図 13 の結果解凍モジュールで所有する鍵によって値を復号化すれば検索結果が表示されるが、属していなければ暗号化されたままの検索結果しか表示されないため、検索が進められない。また実際にデー

タそのものを取得するためにはデータ所有ノードにログインする必要があるため、ノードにアカウントを持っていない場合はグループ内でのみ共有されるデータの検索時にはグ

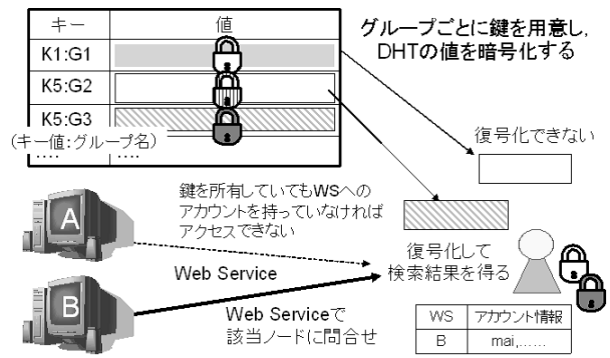


図 14 暗号化によるアクセス制御

ープ名、復号化のための鍵に加えノードへのログインという複数のチェック項目を設けることでセキュリティの強化につながる。

6. まとめと今後の課題

Gfdnavi で扱おうとしているデータは膨大であり、従来のクライアント-サーバ方式ではサーバを設けるコストや帯域の圧迫の問題がある。それらの問題を解消するため Gfdnavi の横断検索機能を P2P で実現しようと考えた。DHT のキーと値のペアを問合せの全てのパターンで登録した場合、ネットワーク上に分散するデータ量の増加が考えられる。その解決策として、まず登録した値を文字列圧縮することでデータ量を減らす方法が挙げられる。さらに最初は上位数件のみを DHT に登録し、ノードに詳細な問合せが生じた場合のみユーザの検索頻度が高いものとして新たに登録することで、データ量を抑えた上で DHT での大まかな横断検索を行うことが可能になると考えられる。また絞り込み操作のためのビット配列はデータ量が多くなり圧縮する必要があるため、variable 同士のパス関係を考慮した上でデータの圧縮を検討している。今後は Gfdnavi への組込みに向け更なる考察を進めていきたい。

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 (課題番号 19024039) 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] 地球流体電脳倶楽部: <http://www.gfd-dennou.org/>
- [2] 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,” 第 18 回データ工学ワークショップ (DEWS2007), D2-8.
- [3] 佐藤麻美, 渡辺知恵美: “P2P を利用した地球物理データのネットワーク横断検索・共有システムの実現に向けて,” 第 18 回データ工学ワークショップ (DEWS2007), D1-9.
- [4] R. Huebsch, J. Hellerstein, N. Lanham, B. T. Loo, S. Shenker, and I. Stoica.: “Querying the Internet with PIER,” In *Proc. the 29th International Conference on Very Large Data Bases*, pp. 321–332, September 2003.
- [5] 首藤一幸, 田中良夫, 関口智嗣: “オーバレイ構築ツールキット

Overlay Weaver, "情報処理学会論文誌:コンピューティングシステム, Vol.47, No.SIG12(ACS 15), pp.358-367, September 2006.

- [6] 重川美咲子, 西澤誠也, 堀之内武, 渡辺知恵美: " Gfdnavi: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 データ属性の探索的検索を利用する検索, " 第 19 回データ工学ワークショップ (DEWS2008).(投稿中)
- [7] Ruby Java Bridge: <http://rjb.rubyforge.org/>