

## 広域センサネットワークの運用構造と多属性検索

落合 秀也<sup>†a)</sup>      松浦 知史<sup>††b)</sup>      砂原 秀樹<sup>††c)</sup>      中山 雅哉<sup>†d)</sup>  
江崎 浩<sup>†e)</sup>

Operating Architecture and Multi-Attribute Search for Wide Area Sensor Networks

Hideya OCHIAI<sup>†a)</sup>, Satoshi MATSUURA<sup>††b)</sup>, Hideki SUNAHARA<sup>††c)</sup>, Masaya  
NAKAYAMA<sup>†d)</sup>, and Hiroshi ESAKI<sup>†e)</sup>

あらまし センサによる環境データ収集は組織単位で独立に運用される。本研究は、これらの組織間で環境データの再利用を促進する広域センサネットワークを実現するため、その運用アーキテクチャと多属性検索機能の導入を提案する。本研究では、まず標準化の方法で組織間での相互接続性を確立する。データ公開に関するアクセスコントロールの実現要求を想定し、データを各自組織のローカルサーバで保管する状況を設定する。この状況下で、分散インデックスを用いたセンサの多属性検索機能を導入する。属性群の定義を運用段階で管理し、柔軟にシステム全体へ反映させる手法もあわせて提案する。提案システムでは、各組織のサーバノードで構成されるネットワークは階層的に運用される。本研究では、サーバノード間の相互接続性、データ提供先のアクセスコントロール機能、種々検索キーによるセンサ検索実現、の観点でシステム検証を行い、これらを同時に実現可能なことを確認した。2007年11月現在、本提案システムはLive E!広域センサネットワークとして、4台のサーバノードで運用されている。

キーワード 広域センサネットワーク、運用アーキテクチャ、相互接続性、多属性検索、データ提供アクセスコントロール

### 1. はじめに

近年、組み込みコンピュータとインターネットの融合により、小型な設備で遠隔地の環境情報を観測することが可能になった。いまや多くのセンサがインターネットに接続され、結果として広域センサ網ができつつある。ところが実際には、これらのセンサシステムは運用組織ごと独自の設計で開発されているのが現状で、多くのセンサデータがオンラインなのにもかかわらず、その一部しか利用できない状況にある。それぞれのセンサデータは、気象観測、災害対策、農業など多くの場面で再利用できる可能性を秘めているが、これを実現するには、データ共有を妨げている上記の現

状を打開する必要がある。一方で、実際的な広域センサネットワーク運用においては、データ提供先のアクセスコントロールが要求される場合がある他、広域に分散したセンサデータを種々観点から発見するための多属性検索を実現する必要もある。本研究では、このような状況を踏まえた広域センサネットワークのアーキテクチャを提案する。

本研究で想定するセンサシステムは、遠隔地のセンサの観測データが、インターネット等を介して、指定されたサーバに定期的に通知される方式のものである。センサ設置および運用は、複数の組織によって独立に行われ、それぞれの組織でインターネットに接続されたデータ収集サーバを運用している。

それぞれの組織のサーバが異なるスキームで独自に構築されている場合、これらを相互接続するためにBrokerを導入する方法がよく知られている(MediatorもしくはAgentと呼ばれることもある)。この方式では、アクセスインタフェースやセンサデータの表現を統一するため、Brokerと各サーバ間にWrapperを作成する。Wrapperは、作成コストの他、対象サーバ

<sup>†</sup> 東京大学

The University of Tokyo

<sup>††</sup> 奈良先端科学技術大学院大学

Nara Institute of Science and Technology

a) E-mail: jo2lxq@hongo.wide.ad.jp

b) E-mail: sato-mat@is.naist.jp

c) E-mail: suna@wide.ad.jp

d) E-mail: nakayama@nc.u-tokyo.ac.jp

e) E-mail: hiroshi@wide.ad.jp

に不足するデータ項目を付け足すなどの操作を伴うことがあり、運用コストの点で実際のでない。Peter Wegner [1] の分類における標準化方式でサーバ間の相互接続性を実現すれば、この問題は解消できる。本研究でもそのアプローチを取る。

実際のシステムを構築する上では、組織 A,B,C にはデータ提供するが、組織 D,E にはデータ提供しないといったアクセスコントロールが必要になる状況も考慮に入れなければならない。それぞれのサーバで自組織のデータを管理し、ここでユーザ認証(組織認証)を行い、オンデマンドにデータ提供を行う形態を取れば、アクセスコントロールを現実的なコストで運用できることは広く知られている。センサ運用者から離れたところにデータを集めて管理する広域センサネットワーク研究が多い中で、本研究では、アクセスコントロールを実現する要求を想定し、センサ運用者からデータを離さずに管理する状況を設定する。

本研究では、センサをプロファイル部とデータ部で表現する。センサには固有の ID を割り当て、プロファイルおよびデータは、この ID で結び付けられる。プロファイル部には、センサに関する各種属性 *profile(n)* (e.g.,  $n$  は、設置場所の緯度経度、住所、管理者名、センサ機種、設置環境、移動体名、関連河川名など) が記載され、ユーザアプリケーションは、この属性を参照することで、センサの分類や各種データ加工に利用する。データ部は、センサが通知してきたデータ集合 *data(t)* であり、最新値データから過去のアーカイブまで、理論的には、すべてが時刻  $t$  を指定することで読み出し可能であるとする。

センサ検索においては、一般に利用形態にあわせて、複数の属性が独立に検索キーとなる。地表面モデル(経緯度モデル)上での検索、特定の会社に属するセンサの検索(組織名検索)の他、移動体名や河川名を検索キーとするような状況が考えられる。このような機能を実現する検索を多属性検索と呼ぶ。

本研究ではアクセスコントロールに関する要求や多属性検索を実現するため、センサデータを運用組織のローカルサーバで管理させた上で、SiteTable と IndexTable を用いた分散環境上の検索手法を提案する。検索キーの属性は、文字列型の場合もあれば、浮動小数点数型である場合もある。緯度、経度、河川などの属性群(スキーマ)は、システムで唯一定義され、全体で共有されていなければならない。これが定期的に更新されるような状況も想定したシステムとして提

案する。

以下、本論文の構成を示す。第 2 章で関連研究を述べ、第 3 章にて Live E!システムデザインを述べる。第 4 章で、実装と運用状況に言及し、第 5 章で評価を行い、第 6 章で考察し、第 7 章で結論を述べる。

## 2. 関連研究

MetBroker [2] は、複数の組織で独立に構築されたセンサデータベースを統合し、統一的なインタフェースを、ユーザアプリケーションに提供する。各組織のデータベースと、Broker との間にそれぞれ Wrapper をプログラムすることで、アクセスインタフェースやデータ表現を統一する。Wrapper の作成・維持は人為的に行われるため、システム管理にかかるコストが規模の増加と共に増大する。

GSN [3] は、仮想化と呼ばれる手法でセンサを標準形で表現し、サーバ間の相互接続性を実現している。明確には述べられていないが、Mill [4] や IrisNet [5] も、センサデータがシステム内で標準化された状況を想定している。Mill は Chord [6] と同様のアーキテクチャを持ち、Z-ordering によって生成される 2 次元 ID 空間を管理することで地理位置検索に関してスケール性を持たせている。ただし、本研究が想定するアクセスコントロールや位置以外を検索キーにする場合については、明確に述べられていない。IrisNet は、OA により地理的な住所構成の木を作成し、その上にセンサデータを対応づける。XPath による検索を実現し、位置以外のもも検索キーに含むことが可能だが、SA からのデータを複数の OA に通知する状況が想定されており、アクセスコントロールをどのように設定可能なのかが明確に述べられていない。本研究では、センサ運用組織にデータ公開権限を持たせ、センサ運用組織のポリシーでアクセスコントロール設定を行える状況を想定している。

Intentional Naming System(INS) [7] は、こちらはフラットなネットワーク構造を持ち、複数の属性を階層的に持つことが可能な Name-specifier (i.e., 検索キー) でサービス検索ができる。サーバが発行する Name-specifier はネットワーク全体に広告され、サーバへの最短経路は分散 Bellman-Ford 法 [8] で作成する。各ノードでのルックアップ処理は Name-specifier の木構造を生かしたインデックスにより達成される。一方、Live E! は、階層的なネットワーク構造を持ち、並列な属性群で構成されるセンサプロファイル上の検

索を行う。センサプロファイルを上位ネットワーク階層へ広告することで経路構築が可能で、検索インデックスは(属性名, 属性値)のペアに対して作成される。

Hyo-Sang Lim et al [9] によると、本研究で対象としているシステムは、Query-initiative 型に分類される。予め分散的に蓄えられているデータに対して、クエリを発行し、データを取得する方式である。一方、予めクエリを発行しておき、発生するセンサデータを能動的にイベント情報として購読者に配信する Data-initiative 型のシステムもある。GSN [3] や [10] [11] は、このような型のシステムを研究している。

### 3. Live E!システムデザイン

Live E!プロジェクト [12] では、本提案システムをデザインするまでの、2005年5月～2007年5月の2年間に渡り、実際の広域センサネットワークへの要求条件を調査した。調査結果のうち、本研究で取り組む要求条件を下記に挙げる。

#### a) 標準形式による相互接続

相互接続の対象となる組織名やサービス場所を設定ファイルに数行書込めば、自律的にコントロール情報が交換され、数時間で、センサ検索やデータ共有を可能にする。

#### b) 組織単位のセンサプロファイル管理

単一組織が管理するセンサは100台程度で、各センサに設定されるプロファイルは10項目程度である。プロファイルが更新される頻度は、経験上平均1.0[回/年・センサ]以下である。

#### c) データ提供先のアクセスコントロール

#### d) 分散環境におけるセンサ検索

#### e) 複数の検索キーの同時サポート

検索キーがA,B,Cとあったときに、Aに関する条件でも、Bに関する条件でも、Cに関する条件でも検索を可能にする。検索キーのデータ型は、少なくとも、数値と文字列には対応しなければならない。比較演算としては、 $<$ ,  $>$ ,  $=$ ,  $<=$ ,  $>=$  があれば、実際的な環境では経験上十分である。

#### f) プロファイルスキーマの柔軟な更新

プロファイルスキーマは、プロファイルの属性名や属性型などの定義で、プロファイル記法および広域検索(分散環境におけるセンサ検索)のクエリ記法、検索処理を規定する。スキーマは、配布ソフトウェアに暗黙的に書込まれるべきではなく、運用中に新しいタイプのアプリケーションが登場する状況を想定し、定

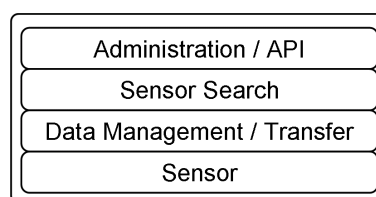


図1 Live E! 広域センサネットワーク・層アーキテクチャ  
Fig. 1 Architecture of Live E! wide area sensor network

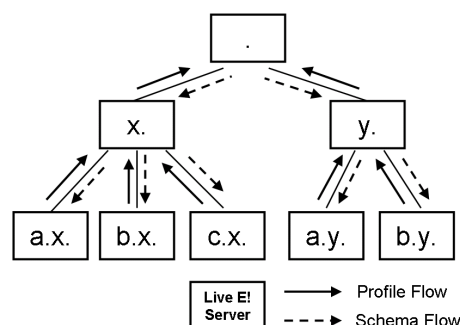


図2 階層構造による Live E!運用ネットワーク  
Fig. 2 Hierarchical operation of Live E! network

義を動的に変更できる必要がある。新規にプロファイル属性を追加する場合はこのような状況に相当する。本研究が対象とする運用形態においては、経験上、スキーマの動的更新が1日以内に達成できれば十分である。

本研究の段階では、実展開/実運用を行う上で上記の課題について解決することに焦点を当てている。スケール性、遅延に関する目標の設定やデザインの提案は将来的な課題である。

#### 3.1 アーキテクチャ

Live E!広域センサネットワークとして、図1に示す層アーキテクチャを提案する。システム全体としてこの層構造を持つが、サーバ単独でもこの層構造を持っている。サーバ間はインターネット上のオーバーレイとして結合され、ネットワークを形成する。結合されたリンクの上で、コントロール情報が交換され、全体として一つのシステムとして協調動作する。

ネットワーク・トポロジは、図2に示すような階層構造になるように、管理層で静的に設定するものとする。静的なのは運用時に計画的なトポロジ設計を行えるようにするためである。サーバにはDNSと同じ Variable-Depth 木 [13] で名前を与える。トポ

ロジに階層構造を採用した理由には、多くの組織の構造が階層的であること、プロフィールスキーマの配信が容易なこと、クエリのルーティングプロトコルが簡素になる<sup>(注1)</sup>ことがある。サーバは DNS と同じ Master/Slave 方式で冗長構成をとることが可能である。

### 3.2 管理/API 層

管理層では、システム全体でプロフィールスキーマを一つ管理し、ルートサーバから配信している。具体的には、各サーバが上位階層サーバのスキーマを定期的に読み込むことで、ルートサーバによって発行されたスキーマを、システム全体に行渡らせる。このようにすることで、プロフィールスキーマの柔軟な更新を可能にする。

各センサ運用組織の管理層では、下記の設定を行うことができる。

- 近隣ノードとのリンク設定
- センサプロフィールの発行
- データ提供先アクセスコントロール設定
- 各種パラメータ (キャッシュ時間など) の設定
- 基本設定 (ログ出力先など)

センサは、各組織で設置され運用されることを考えると、プロフィールは組織ごとに発行することが妥当である。ここでプロフィールのいくつかの属性は、自動生成することも可能だが、全属性の設定の自動化は難しいため、人の手で設定できる必要がある。GPS により経緯度情報は自動化できるし、センサ機器自身が機種情報を持つことでの自動化もできる。しかし、どのような設置環境 (e.g., 路面, 公園) かや、移動体名などの情報は人手による設定が実際的である。更新頻度は経験上 1.0[回/年・センサ] 程度と予測されるため、人の手による運用でも十分対応可能である。

アクセスコントロール設定は、組織のセンサ公開ポリシーを設定するものである。本提案システムでは、検索要求およびデータ取得要求は各サーバが発行し、発行組織名 (サーバ名) を許可リストや拒否リストに追加することで設定する。図 2 において、`<*.x.>` を許可リストに加えた場合、`{<a.x.>, <b.x.>, <c.x.>, <x.>}` からのデータ取得要求は許可される。アクセスコントロール設定は、データ転送層の Reference Monitor [14] に

(注1): 階層的なネットワークであれば、プロフィール情報を上位階層へ広告すればよい。一方、一般的な (i.e., フラットな) ネットワークでは、プロフィール情報を全体で共有する必要があり、さらには経路にループができないようにしなければならない。

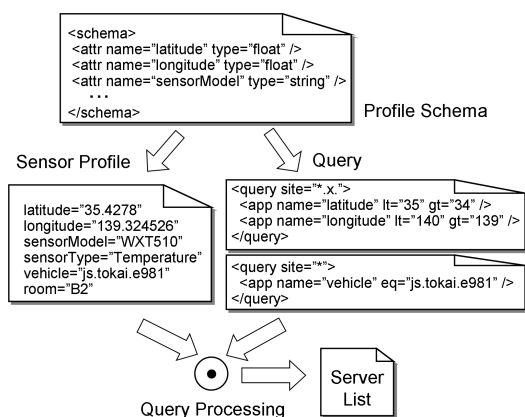


図 3 スキーマ、プロフィール、クエリ、検索プロセスの関係

Fig. 3 Schema, Profile, Query, Processing

対して行われ、センサ検索後のオンデマンドなデータ取得要求に対して働く。

API 層は、ユーザアプリケーションがセンサ検索/データ読み出しを行うためのインタフェースを提供する。全センサの最新値データの取得、特定の領域の最新値データの取得などのインタフェースを提供する。ユーザリクエストを受信してから、分散環境の中で検索を実行し、データを読み出すまで、本研究の段階では、単一の Remote Procedure Call (RPC) の中で行うように API をデザインしている。

### 3.3 センサ検索層

センサ検索層では、管理層で設定された Live E! ネットワーク上での、センサ広域検索を実現する。下位階層のサーバノードのプロフィールを定期的に読み込み、検索クエリのルーティングを自律的に実現する。

図 3 にプロフィールスキーマ、プロフィール、検索クエリ、検索プロセスの関係を示す。検索プロセスは、検索クエリに合致するプロフィールを含むサーバをリストアップする処理である。スキーマは、プロフィール属性名、属性型、取りうる値などを規定するもので、管理層から与えられる。各センサ運用組織は、センサに対して、このスキーマに適合するプロフィールを発行し、ローカルサーバに登録してある。検索クエリもスキーマに適合する形で作成され、検索プロセスに利用される。具体的には、検索条件が  $34 < \text{latitude} < 35 \wedge 139 < \text{longitude} < 140$  であるような検索クエリを発行することが可能で、検索対象とするサーバ (組織名) を限定することも可能である (図 3 の例では、site で検

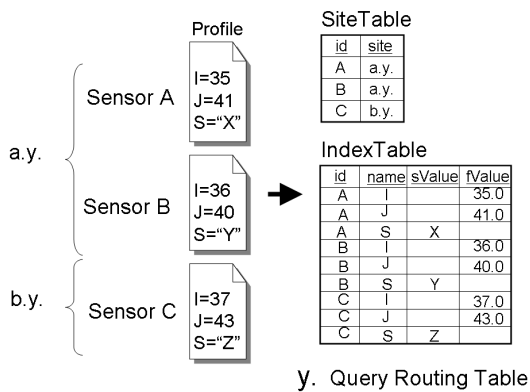


図 4 クエリ経路表の構築

Fig. 4 Construction of query routing table

索対象とするサーバを限定している)。組織名検索は DNS 同様に Patricia-Trie アルゴリズム [15] で達成される。

本研究では、分散インデックスを用意し、検索クエリをルーティングすることで、検索プロセスを分散環境上で実現する形態をとる [16]。検索を、近隣ノードへのクエリ転送処理の連続形に、抽象化することで、分散環境上での検索プロセスの実装を将来的に交換しやすくするためである。本研究では、クエリ問合せ先ノードが、応答としてリダイレクション (i.e., 次に問い合わせるべきノードの情報) を発行することを転送と呼び、転送を連続的に行うことでネットワーク上での検索を実現させることをルーティングと呼ぶ。本研究では、分散インデックスをクエリ経路表と呼ぶ。検索クエリのルーティングを自律的に実現するために、プロフィールを上位サーバノードが定期的に取り込んでいる。

INS [7] では、Name-specifier の木構造を生かしたインデックスを作成するが、本研究のクエリ経路表では、(属性名, 属性値) のペアに対して、インデックスを作成している。以下、クエリ経路表の作成方法とクエリ転送処理について解説する。(ここではサーバのことをノードと表現する)。

クエリ経路表の作成 図 2 において、すべてのノードはクエリ経路表を保持する。この経路表の作成は、子ノードの保有する (そのすべての子孫も含めた) プロファイル情報を読み込んだ上で行われる。<x.>であれば、{<a.x.>, <b.x.>, <c.x.>} からプロフィール情報を読み込み、経路表が作成される。<y.>であれば、{<x.>, <y.>} からプロフィール情報を読み込み、作成

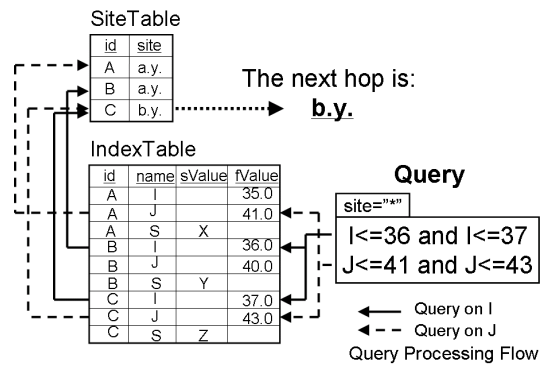


図 5 クエリの転送処理

Fig. 5 Query forwarding process

される (このとき、子孫である {<a.x.>, ..., <b.y.>} のプロフィール情報も {<x.>, <y.>} 経由で、<y.>に届いている)。

図 4 にノード <y.>でのクエリ経路表の作成手順を示す。<y.>には、子ノード {<a.y.>, <b.y.>} があり、<a.y.>およびその子孫にあたるノードにはセンサ A と B、<b.y.>およびその子孫にあたるノードにはセンサ C が存在している。それぞれのセンサプロフィールは、図 4 に示す通りとする。ここでプロフィールの属性は、I および J (e.g., 緯度, 経度) は浮動小数点数型、S が文字列型 (e.g., センサ機種) とするようプロフィールスキーマで規定されている。<y.>は、<a.y.>および<b.y.>から、センサプロフィールを受け取り、SiteTable と IndexTable を次のように作成する。SiteTable では、センサ ID と子ノードを対応させ、IndexTable には、それぞれのセンサプロフィールを (センサ ID, 属性名, 文字列型の値, 浮動小数点数型の値) で展開する。図の例では、I および J は浮動小数点数なので、fValue に値が格納され、S は文字列型なので、sValue に値が格納されている。次に述べる転送処理のため、(name, sValue) および (name, fValue), それぞれに関して、検索インデックスを作成してある。IndexTable, SiteTable は、定期的に (e.g., 10 分おきに) 上記の操作が実行されることで更新され、最新の状態に保たれる。

クエリ転送処理 すべてのノードはクエリ経路表を保持しており、クエリ問い合わせがあると、応答として次に問い合わせるべきノード情報 (i.e., 転送先 IP アドレス等) を返す。クエリは、検索対象となるノード名の範囲と、複数の属性に関する条件式で構成され

る。転送処理は、次のようになる。まず、(1) 検索クエリを、属性ごとの条件文に分割し、IndexTable において、それぞれ条件に適合するセンサ ID を考える。(2) SiteTable において、それらセンサ ID の積集合を取り、最終的に残ったセンサ ID を持つ子ノードをクエリの転送先として算出する。

具体例として、図 4 で示した、<y.>で作成された経路表に基づき解説する。図 5 に示す意味の検索クエリが発行されたとする。site が、検索対象となるノード名の範囲で、I および J が属性に関する条件式である。(1) まず検索クエリを、条件 I と条件 J に分割し、IndexTable において適合するセンサ ID を、それぞれ {B,C}, {A,C} と算出する。(2) SiteTable にてセンサ ID の積集合を取ると C のみが残るので、<b.y.>がクエリの転送先として導出される。

この方式により、プロフィール属性を検索キーにしたノード検索が可能になる。検索対象領域が狭ければ、クエリ転送処理は多くの場合  $O(\log N)$  以下で達成できる ( $N$  はセンサ数、検索対象領域が狭い状況ではインデックス検索コストが支配的)。この方式自体では、ノードの経路表は  $O(N)$  で増加する。経路表の作成コストは、検索インデックスが B-Tree [17] で実装されていれば、 $O(N \log N)$  に従う。

センサ数の増大に伴い、上位ノードの経路表が増大してくる。結果、(1) 定期的に行われるクエリ経路表の作成処理時間が増大する。同時に (2) クエリで指定された検索対象領域が広い場合、転送処理時間が増大する。スケール性を実現する研究は将来的な課題とし、本研究の段階では、検索の機能性を犠牲にすることでこの問題に対処することを考えている。ルートサーバから途中の階層まで組織名(サーバノード名)をキーとした検索を行なわせ、その先から、他のキーでの検索を行わせる。一定規模以上になった場合、上位ノードでの SiteTable, IndexTable の作成を諦めるか、検索対象領域が広いクエリの転送処理を禁止する。提案システムでは、組織名をキーにした検索では、クエリ転送処理時間は事実上  $O(1)$  になる(理由: SiteTable, IndexTable での処理が不要なため)。

検索の実装 [18] には、Iterative 方式を採用している。Iterative 方式は、Recursive 方式と比較しセッション時間が短く、上位ノードの負荷を小さく抑えることが可能であると共に、タイムアウト閾値を低く抑えることが可能なためである。

### 3.4 データ管理/転送層, センサ層

データ管理層では、センサ層で認証されたセンサデータを扱う。センサから受け付けたデータは、まず最新値データとして扱われ、アグリゲーション (e.g., MAX, MIN, AVERAGE 演算) 用に展開され、オンラインに読み出せる形で保存される。ローカルサーバ内にデータを配置し、Reference Monitor [14] をその前に設置することでアクセスコントロールを行う。Reference Monitor は、管理層での設定に従って、要求発行元のサーバにデータを提供すべきかどうか判断する。要求発行元のサーバ識別はデータ取得要求時に、自サーバ名を同時に提示させることで行う。本研究の対象外だが、要求身元の確認(ペリファイ)や改竄を防ぐ機構などはここに組み込まれる。

データ転送層では、センサ検索層でリストアップされたサーバそれぞれに対して、データ取得要求を発行し、その結果取得できたデータをマージする。ここで、データ取得要求は、最新値データ、過去データ、アグリゲーションされたデータの読出し要求に分類され、その分類の中でも、時刻範囲や経緯度範囲など、選択的にそのサーバノードが管理しているデータを指定できる。

センサ層では、センサとの通信機能を提供し、センサ認証も行う。データ通知インタフェースは、システム内で標準化されており、そのインタフェースに適合したセンサがデータを通知してくる状況を想定している。

## 4. 実装/運用状況

2007 年 11 月現在、Live E! プロジェクトで展開を行っている典型的な Live E!サーバは、Java2 ver.1.5. で実装され、その規模はクラス数にして 291 個である。データベースに PostgreSQL を使用し、クエリ経路表はデータベースのテーブルとして実装している。サーバ間通信は、SOAP で行われ、交換されるメッセージは XML で記述されている。SOAP エンジンの実装には、Axis1.4 を使用している。

このシステムは、2007 年 8 月から運用を開始した。図 6 に 2007 年 11 月現在の運用状況を示す。

ルートサーバ<.、<wrapper.>, <jp.>は、東京大学に設置されており、それぞれ 0 個、148 個、3 個のセンサを管理している。<nara.jp.>は、奈良先端科学技術大学院大学に設置され、2 個のセンサを管理している。サーバ間の Round Trip Time(RTT) は、最大 10ms

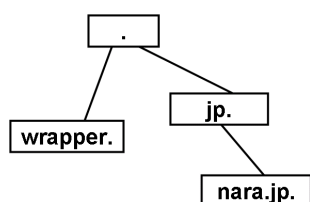


図 6 2007 年 11 月現在の運用状況  
Fig. 6 The current operational status in Nov. 2007.

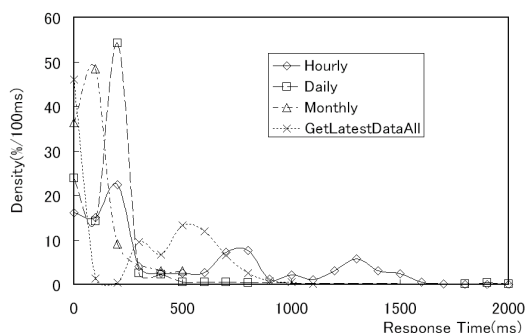


図 7 ユーザリクエスト処理時間分布  
Fig. 7 Distribution of user request processing time

程度である。

図 7 に、<jp.>で実際に観測された「ユーザリクエストの処理時間分布」を示す。Hourly, Daily, Monthly, GetLatestDataAll は、それぞれ、(1) 時間単位、(2) 日単位、(3) 月単位でアグリゲートされたデータと (4) 最新データ、の読出しリクエストを意味する。リクエストごと扱うデータ量が異なるため、単純な比較はできないが、この分布から処理時間の期待値を計算したところ、次のようになった。

- Hourly - 478ms
- Daily - 178ms
- Monthly - 94ms
- GetLatestDataAll - 283ms

ユーザリクエストの平均処理時間が 100ms ~ 500ms 程度で運用されていることがわかる。

図 8 に、「ルートサーバにおけるクエリ転送処理時間分布」を月ごとに示す。クエリ転送は 2 ~ 4ms でほぼ完了している。各月の総クエリ数は下記であった。

- 8 月 1026 回
- 9 月 814 回
- 10 月 3500 回
- 11 月 3992 回

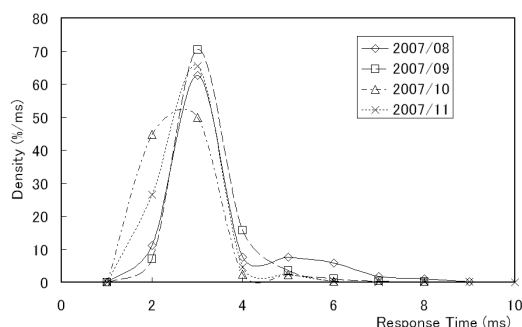


図 8 ルートサーバでのクエリ転送処理時間分布  
Fig. 8 Distribution of query forwarding processing time at the root server

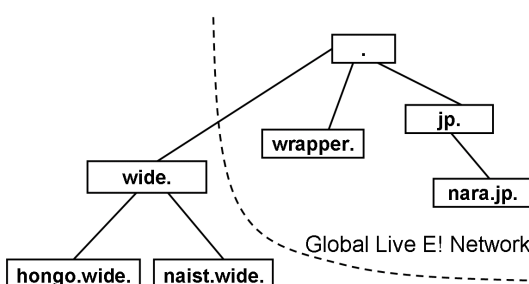


図 9 "wide" の Global Live E! Network への参加実験  
Fig. 9 "wide" is joining to the Global Live E! Network

## 5. システム評価

### 5.1 相互接続性と運用管理

センサ運用組織 wide と、その子組織 hongo, naist が独自にセンサを管理運営しており、Live E! の他組織とのデータ共有を行う状況を想定する。それぞれの組織で Live E!サーバをインストールし、図 9 のように、グローバル Live E!ネットワークに接続した (管理層で各リンクを設定した)。動作開始の数秒後、リンクの接続状態がアクティブになり、グローバル Live E!ネットワークで管理されているプロファイルスキーマが、これら 3 台のサーバに配布される様子が観測された。上位サーバからのスキーマ読み込み周期を、300 秒 (=5 分) に設定した場合、定常動作状態においては、<>で更新されたスキーマは、10 分以内に、<hongo.wide.> や <naist.wide.> へ到達した。<>のスキーマに更新があった 5 分後に、<wide.> がそのスキーマを読み込み、<hongo.wide.> と <naist.wide.> が、さらにその 5 分後に、<wide.> からスキーマを読み込んだケースが到達時間が最大になる場合で、10 分であった。

こうして構築されたネットワーク上で、それぞれの組織の管理下で、独立にセンサ登録やプロフィール管理を行えることを確認した。登録したプロフィールは、上位のサーバに伝播することも確認した。伝播周期を300秒に設定した結果、<hongo.wide.>に登録したセンサのプロファイルは、10分以内に<.>にまで到達することが確かめられた。ここで10分間は、実運用上ほとんど問題にならない時間だと考えている。従来型方式の場合、相互接続のための Wrapper 作成に、日単位での時間を要していた。つまり、 $10^{-4}$  程度の時間短縮を実現できたことになる。

## 5.2 検 索

次の状況を想定し、組織名、経緯度、移動体名、河川名、それぞれをキーとした検索が可能であることを確認した。(1)hongoは、東日本エリアを重点的にセンサ展開する。河川情報に強く関係した水位センサや、列車に取り付けられたセンサも管理している。(2)naistは、西日本エリアを重点的にセンサ展開する。水位センサや、列車のセンサも管理している。(3)wideは、hongoやnaistに属しないwideメンバが持つセンサを管理する。日本全体にセンサを展開するが、河川関連のセンサや、列車についてのセンサも管理している。

サーバのトポロジ上では、センサは組織で分類されているが、経緯度、移動体、河川は、複数の組織にまたがって存在するといった現実起こりうる状況を再現している。例えば、hongoとwideは、*river=jp.arakawa*に関連するセンサをそれぞれで持っている。

以上の状況を想定し、図10のようにプロフィール登録した。検索実験と結果を、表5.2に示す。この結果からわかるように、複数の異なる検索キーで、それぞれ検索可能であった。

## 5.3 センサデータの公開制御

図9のLive E!ネットワークトポロジにおいて、wide(hongo, naistを含む)の特定のセンサデータは、wide内だけで閉じて、外部に公開すべきでない状況を考える。hongoで収集しているデータは、wide内での共有により組織全体の活動に貢献できるが、不特定多数の外部組織への公開は不利益を生じうる場合である。

hongoにおいて、このセンサに対するデータ公開ポリシーを<\*.wide.>のみに設定した。結果、外側の組織<wrapper.>、<jp.>、<nara.jp.>からの<hongo.wide.>のデータ読出しは不可能となり、<wide.>、<naist.wide.>、<hongo.wide.>からのみ読出

し可能になった。こうして目的のアクセスコントロールを実現できることが確かめられた。

## 6. 考察と課題

Wrapperをプログラムして複数のデータベース間の相互接続性を確立する方式と比べ、本研究の方式(i.e., データ表現やアクセスインタフェースを標準化し、自律分散的に広域検索を実現させる)では、相互接続に関するコストを(時間にして $10^{-4}$ 程度に)削減できた。これはネットワークのトポロジに柔軟性を与えることが可能になったことを示唆する。実際の運用にあたっては、トポロジ設計を計画的に行う必要がある。国ごと地域ごとに運用サーバを計画的に配置することで、検索に伴う分岐が減少し、検索効率を上げることが可能になると考えられる。計画的な割り当ては、システム全体のパフォーマンスに左右するとする見解はRFC2050[19]でも述べられている。

計画的な運用には、プロフィールスキーマ管理も挙げることができる。プロフィールスキーマを発行するルートサーバの役割は、インターネットでのInternet Assigned Numbers Authority(IANA)[20]に相当する。スキーマを動的に配信する仕組みを導入したことで、システム全体に渡ったプロフィール記法や検索キーの管理を可能にしたが、具体的なプロフィールスキーマのモデル設定・管理方法は運用上の課題である。

本研究では、種々検索キーでの広域検索を実現するために、センサプロフィールを上位サーバノードに読み込ませることで、検索クエリを広域にわたってルーティングさせる手法を提案し、一定の規模においては実現可能であることが明らかとなった。本研究で扱った規模であれば、集中インデックスサーバを設置し、そこに問い合わせるだけでクエリに合致するすべてのノードを応答する検索方式も可能かもしれない。あえてクエリ・ルーティングにより検索を実現する手法[16]を採用したのは、将来的に様々なルーティング実装を組み込み可能にするためである。

本研究の段階では、移動体は、移動体名をキーにすることで検索をする利用シナリオを考えていた。これは、刻々と変化する経緯度をプロフィールとして扱うと、ネットワーク内でのプロフィール伝播遅延問題を扱う必要が出てくるためである。開発したシステムでは、経緯度を含め、あらゆる属性は、プロフィールとしてもデータとしても登録できるようになっており、実際の運用では移動体の刻々と変化する経緯度情報



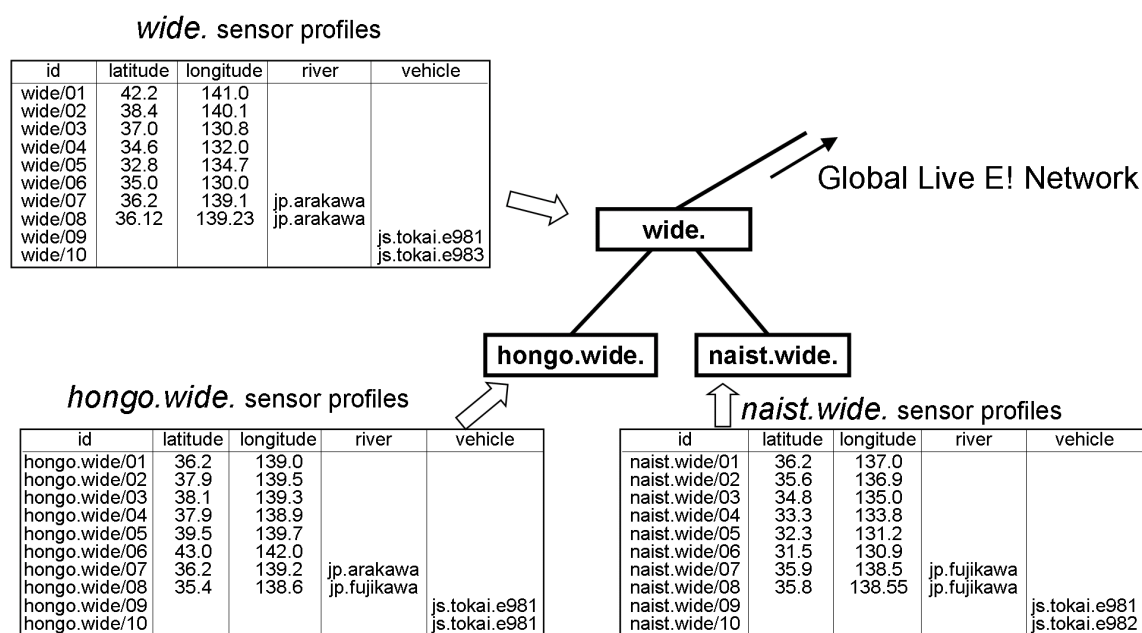


図 10 組織名，緯度経度，河川名，移動体名による検索実験環境  
 Fig.10 Search experiment on organization names, latitude and longitude, river names and vehicle names

表 1 検索実験結果  
 Table 1 The result of the search experiment

| 検索キー    | 条件                           | 平均検索時間 (ms) | 結果   |
|---------|------------------------------|-------------|--|
| 組織名     | *                            | 354         | ., jp., nara.jp, wrapper., wide., naist.wide., hongo.wide. |
|         | *.wide.                      | 167         | wide., naist.wide., hongo.wide.,                           |
|         | naist.wide.                  | 58          | naist.wide.,   |
| 経緯度     | [36,40] × [139,140]          | 248         | wrapper., wide., hongo.wide.,                              |
|         | [30,35] × [130,133]          | 224         | wrapper., wide., naist.wide.,                              |
| 河川名     | jp.arakawa                   | 157         | wide., hongo.wide.,  |
|         | jp.fujikawa                  | 234         | naist.wide., hongo.wide.,                                  |
| 移動体名    | jp.tokai.e981                | 208         | wide., naist.wide., hongo.wide.,                           |
|         | jp.tokai.e982                | 149         | naist.wide.,   |
|         | jp.tokai.e983                | 118         | wide.,   |
| 組織名+経緯度 | *.wide.+ [36,40] × [139,140] | 119         | wide., hongo.wide.,  |

は、データとして扱うことになる。経緯度をプロフィールとしても登録し、経緯度による検索で見えるようにするには、プロフィールの伝播遅延等の課題を今後扱わなければならない。

## 7. おわりに

本研究では、インターネット上のセンサデータを自律分散的に共有する Live E!広域センサネットワークの提案を行った。データ表現やアクセスインタフェースの標準化により、データ共有にかかる運用コストの削減(分単位でのネットワーク構築)を可能にした。

本研究では、データ提供先のアクセスコントロールに関する要求を考慮し、それぞれのセンサ運用組織のローカルサーバでデータを保管する状況を考えて、その上で、プロフィールを広域で共有させることで、種々検索キー (i.e., 組織名, 緯度経度, 河川名, 移動体名) によるセンサ検索を分散的に実現した。検索には、SiteTable と IndexTable を組み合わせたクエリ経路表の作成と、クエリ転送処理を伴う方式を提案した。この方式は、センサ数の増大に伴うスケール性に課題が残ると予想され、今後さらなる改良が要求される。プロフィールおよび検索は、システム全体で管理

されるプロファイルスキーマに基づいて規定されるように設計し、柔軟な属性変更が可能になった。センサ運用組織は、データ提供先の許可/禁止リストを作成するだけで、アクセスコントロールを実現できた。

謝辞

本研究は、Live E!プロジェクトの広域センサネットワーク技術開発の一環として実施したものである。総務省の委託研究「ユビキタスネットワーク認証・エージェント技術の研究開発」の成果の一部を活用したものである。

## 文 献

- [1] P. Wegner: "Interoperability", ACM Computing Surveys, **28**, 1, pp. 285-287 (1996).
- [2] S. Ninomiya, T. Kiura, A. Yamakawa, T. Fukatsu, K. Tanaka, H. Meng and M. Hirafuji: "Seamless integration of sensor network and legacy weather databases by metbroker", IEEE/IPSJ SAINT2007 workshop, p. 68 (2007).
- [3] K. Aberer, M. Hauswirth and A. Salehi: "Infrastructure for data processing in large-scale interconnected sensor networks", IEEE MDM 2007 (2007).
- [4] S. Matsuura, K. Fujikawa and H. Sunahara: "Mill: A geographical location oriented overlay network managing data of ubiquitous sensors", IEICE Transactions on Communications, **E90-B**, 10, pp. 2720-2728 (2007).
- [5] P. B. Gibbons, B. Karp, Y. Ke, S. Nath and S. Seshan: "Irisnet: an architecture for a world wide sensor web", IEEE Pervasive Computing, **2**, 4, pp. 22-33 (2003).
- [6] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek and H. Balakrishnan: "Chord: A scalable peer-to-peer lookup service for internet applications", ACM SIGCOMM, pp. 149-160 (2001).
- [7] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan and J. Lillery: "The design and implementation of an intentional naming system", Proceedings of the 7th ACM symposium on Operating systems principles, pp. 186-201 (1999).
- [8] R. Bellman: "On a routing problem", in Quarterly of Applied Mathematics, **16**, 1, pp. 87-90 (1958).
- [9] H.-S. Lim, J.-G. Lee, M.-J. Lee, K.-Y. Whang and I.-Y. Song: "Continuous query processing in data streams using duality of data and queries", ACM SIGMOD (2006).
- [10] D. J. Abadi, W. Lindner, S. Madden and J. Schuler: "An integration framework for sensor networks and data stream management systems", 30th VLDB Conference, pp. 1361-1364 (2004).
- [11] H. Ochiai, Z. Wang, R. Oguchi, A. Sugiyama, Y. Sakamoto, S. Ishida and H. Esaki: "Application of content-based network for sensor data distribution system", IEEE/IPSJ SAINTW'07, p. 74 (2007).
- [12] M. Nakayama, S. Matsuura, H. Esaki and H. Sunahara: "Live E! project: Sensing the earth", LNCS, **4311**, pp. 61-74 (2006).
- [13] P. V. Mockapetris and K. J. Dunlap: "Development of the domain name system", ACM SIGCOMM, pp. 123-133 (1988).
- [14] A. S. Tanenbaum and M. V. Steen: "Distributed Systems", chapter General Issues in Access Control, pp. 414-418, Peason Education, Inc., second edition (2006).
- [15] D. R. Morrison: "PATRICIA - practical algorithm to retrieve information coded in alphanumeric", Journal of the ACM, **15**, 4, pp. 514-534 (1968).
- [16] H. Crespo, A. Garcia-Molina: "Routing indices for peer-to-peer systems", IEEE Distributed Computing Systems, pp. 23-32 (2002).
- [17] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein: "Introduction to Algorithms", chapter 18: B-Trees, pp. 434-454, MIT Press, second edition (2001).
- [18] A. S. Tanenbaum and M. V. Steen: "Distributed Systems", chapter Implementation of Name Resolution, pp. 205-209, Peason Education, Inc., second edition (2006).
- [19] K. Hubbard, M. Kosters, D. Conrad, D. Karrenberg and J. Postel: "RFC2050: internet registry ip allocation guidelines" (1996).
- [20] "Iana". <http://www.iana.org/>.

(平成 xx 年 xx 月 xx 日受付)

### 落合 秀也

平成 18 東京大学・工・電子情報卒。同年 同大学大学院・情報理工学系研究科修士課程。現在に至る。組込みシステム，広域センサネットワークの研究に従事。

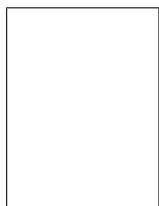
### 松浦 知史

平成 15 立命館大学・理工・物理卒。平成 17 奈良先端科学技術大学院大学・情報科学研究科修士課程了。同年より同大学・情報科学研究科博士課程。オーバーレイネットワーク，センサネットワークの研究に従事。

### 砂原 秀樹

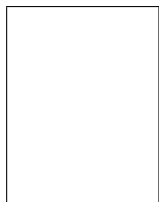
昭和 58 慶大・工・電気卒。昭和 63 同

大学大学院博士課程了。同年 電通大・情報・助手。平成 6 奈良先端科学技術大学院大学・情報科学センター・助教授。平成 13 同大学・情報科学センター・教授。平成 17 同大学・情報科学研究科・教授。現在に至る。工学博士。インターネット、大規模広域分散環境、ネットワーク、並列処理、オペレーティングシステム、電子図書館に関する研究に従事。情報処理学会、電子情報通信学会、ACM、IEEE 各会員。



中山 雅哉

昭和 59 慶大・工・電気卒。平成元 東京大学大学院博士課程了。同年 豊橋技術科学大・知識情報工・助手。平成 5 東京大学・大型計算機センター・助教授。平成 10 同大学大学院・工学系研究科・助教授(兼任)。平成 11 同大学・情報基盤センター・助教授。平成 13 同大学大学院・新領域創成科学研究科・助教授(兼任)。IEEE, 電子情報通信学会, 情報処理学会, 各会員。



江崎 浩

昭和 62 九州大学・工・電子 修士課程了。同年 (株) 東芝 入社。平成 2 米国ニュージャージー州 ベルコア社。平成 6 コロンビア大学・客員研究員。平成 10 東京大学 大型計算機センター・助教授。平成 13 同大学大学院・情報理工学系研究科・助教授。平成 17 同大学大学院・同研究科・教授。現在に至る。工学博士(東京大学)。MPLS-JAPAN 代表, IPv6 普及・高度化推進協議会専務理事, JPNIC 理事, ISOC 理事。

**Abstract** Environment sensing systems must be operated independently by organizations. This paper proposes an operating architecture and multi-attribute search for wide area sensor networks intended to share and reuse those data. Standardization allows interoperability among servers of those organizations. Sensor data must be managed at their origin server to implement access control functionality. Under the situation, we challenge multi-attribute sensor search, with addressing schema administration of the attributes. We evaluate our system in terms of interoperability, access control, and multi-attribute search. Our proposed system works with four server nodes as Live E! wide area sensor network in Nov. 2007.

**Key words** Wide Area Sensor Networks, Operating Architecture, Interoperability, Multi-Attribute Search, Access Control