

### 3.4.4 情報システム連携の枠組み構築

#### (1) 業務の内容

##### (a) 業務の目的

広域連携における情報共有を実現するには、既存システムや新規システムなど多種多様な情報システムを相互に連携させていかなければならない。このためには、情報共有システム連携の枠組みを提供することが不可欠となる。これまで開発してきた減災情報共有プロトコル(MISP)および減災情報共有データベース(DaRuMa)を基盤として、個別研究テーマ(1)④と連携し、広域連携に必要な機能を洗い出し、プロトコルの再設計およびそれに基づいたデータベースプロトタイプシステムの拡張を行う。また、個別研究テーマ(2)①で整理された情報コンテンツを分析し、多数のコンテンツの情報表現を記述するスキーマの作成・管理を行うためのシステムを開発すると同時に、基本的なデータ構造についての設計・標準化を行う。プロジェクト後半に実施される実証実験に向け、プロトコル等の情報共有データベースの枠組みを完成させる。また、実証実験のために必要な接続ツール群を拡充し、情報共有データベースプロトタイプシステムの構築を行って、実証実験に参画することを目的とする。なお、このプロジェクトを通じて開発される各システムはオープンソースの形で順次公開され、一般利用などを推進していく予定である。

##### (b) 平成20年度業務目的

個別研究テーマ(2)①の研究で評価実験のシナリオが行われ、個別研究テーマ(2)②の研究で評価実験のための情報コンテンツが整理される。したがって、評価実験で行われる情報項目毎の共有形態に応じて、既開発の減災情報共有プロトコル(MISP)および減災情報共有データベース(DaRuMa)を基盤として、追加すべき機能を洗い出し、情報表現形式の相互変換に必要な情報処理技術を開発とプロトコルの再設計を行い、それに基づいたデータベースプロトタイプシステムの拡張を行う。

##### (c) 担当者

所属機関	役職	氏名	メールアドレス
独立行政法人産業技術研究所 情報技術研究部門	主任研究員	野田五十樹	
同	テクニカルスタッフ	下羅弘樹	
同	特別研究員	秋山英久	

#### (2) 平成20年度の成果

##### (a) 業務の要約

本年度は、評価実験で行われる情報項目毎の共有形態に応じて、既開発の減災情報共有プロトコル(MISP)および減災情報共有データベース(DaRuMa)を基盤として、追加すべき機能を洗

い出し、情報表現形式の相互変換に必要な情報処理技術を開発とプロトコルの再設計を行い、それに基づいたデータベースプロトタイプシステムの拡張を行った。具体的には以下のような研究開発を行った。

- 他機関で開発が進められている各種システムの接続を可能とするため、データ表現および接続プロトコルの整備を進めた。
- 既存の各種システムや情報について、DaRuMa による情報共有の有効性・効率性確認を行った。

## (b) 業務の成果

本プロジェクトでは、減災情報共有データベースを核として、首都直下地震に対処するための各種情報システムを相互連携させることを目標としている。本サブテーマではこの目標を可能とするために、減災情報共有データベースの1実装である DaRuMa とそこで採用しているプロトコル MISP (減災情報共有プロトコル, Mitigation Information Sharing Protocol) に改良を加え、より実用的なものとするをミッションとしている。このために本年度は、以下のような開発に取り組んだ。

- 東京大学で進められている病院内情報システム、山梨大学で進められている自治体庁内システム、JAXA にて進められている救急ヘリ管制システム、および東京大学で進められている汎用ビューワの接続を可能とするため、データ表現および接続プロトコルの整備を進めた。特に、ビューワで必要とされた検索結果の整列機能に関するプロトコル拡張について、WFS に準拠しつつ独自形式による機能追加を行い、データベースシステムに実装した。
- 既存の各種システムや情報について、DaRuMa による情報共有の有効性・効率性をしめす為、以下のような連携の有効性検証と拡張を進めた。
  - 広域緊急医療情報システム (EMIS) の情報を MISP および DaRuMa を介して再利用可能な形に整理し、GIS により視覚的に状況を表示するシステムの改良を進め、実際に医療に携わる人による評価実験を進め、その有用性を確認した。
  - 道路の利用状況を反映するため、利用状態の情報表現およびそれに対応するスキーマの設計を行った。

以下では、各々の成果を個別に詳述する。

### 1) DaRuMa/MISP の拡張

#### a). 検索機能の拡充

##### i) 検索結果整列機能

これまでの MISP では、検索条件に合致したデータは順不同で出力されることになっていた。従来進めてきた MISP を用いた連携実験では、各々の災害情報の件数は比較的少数であり、順不同で扱っていてもそれほど大きな問題は生じてこなかった。一方、本プロジェクトで本年度取り組んだ救急ヘリの運行情報では、本格運用の際には情報件数は膨大になることが考えられる。また、将来的に自動化されたセンサー情報を扱う場合などを考えると、膨大な検索結果件数の

整理機能を整備する必要が出てくる。特に検索結果になんらかの形で情報に優先順位をつける  
て整列させる機能が必要となる。

この機能をプロトコルに追加するにあたっては、以下のいくつかの形式が考えられる。

- GetFeature あるいは Query タグレベルにおいて出力形式指定の一部として整列条件を属性(Attribute)の形式で記述する。
- UNIX などにおけるパイプラインの考え方にに基づき、Query の構成要素として、Filter により条件指定された検索結果を受け、整列を行うパイプの形式で指定する。

本プロジェクトではこれらのうち、WFS(Web Feature Service)<sup>1)</sup>の Filter Encoding<sup>2)</sup>でも採用されている後者の考え方でプロトコルの拡張を行った。

具体的なプロトコルの形式は以下のとおりである。

```
<GetFeature xmlns="http://www.infosharp.org/misp">
  <Query typeName="name of feature">
    <Filter>
      Any Condition
    </Filter>
    [<SortBy>
      [<SortProperty>
        <PropertyName>property for the key of sort </PropertyName>
        [<SortOrder>ASC|DESC</SortOrder>]
      </SortProperty>]*
    </SortBy>]
  </Query>
</GetFeature>
```

図 1 検索結果整列機能の要求プロトコル形式

このうち、SortBy 要素以下が新たに追加された部分であり、その中の SortProperty 要素内の  
PropertyName 要素に、整列の際に参照するデータ要素を XPath の形式で指定する。また、  
SortOrder 要素には並び順の指定として、昇順(ASC)、降順(DESC)のどちらを使うかを指定でき  
る。また、SortProperty は複数個指定することができる。この場合、与えられた SortProperty  
の順で整列の優先順位が定められ、より優先順位の高い SortProperty のデータ要素  
(PropertyName)の値が同値であった場合に、それ以降の SortProperty で指定されたデータ要素  
により整列を行う。指定されたすべてのデータ要素の値が同値である場合は、その整列結果は  
順不同である。

具体的な例(図 2、図 3)は以下のとおりである。この例では、2009 年 3 月 10 日以降に観測  
された河川の水位情報について、観測所・観測時刻・観測水位の情報を観測水位で降順に整列  
して結果出力するものである。

```

<GetFeature xmlns="http://www.infosharp.org/misp"
  xmlns:rev="http://www.infosharp.org/test/river">
  <Query typeName="rev:RiverInformation">
    <Filter>
      <PropertyIsGreaterThan>
        <PropertyName>rev:timestamp</PropertyName>
        <Literal>2009-03-10T00:00:00+09:00</Literal>
      </PropertyIsGreaterThan>
    </Filter>
    <SortBy>
      <SortProperty>
        <PropertyName>rev:waterLevel</PropertyName>
        <SortOrder>DESC</SortOrder>
      </SortProperty>
    </SortBy>
  </Query>
</GetFeature>

```

図 2 検索結果整列機能の要求例

```

<misp:GetFeatureResponse
  xmlns:misp="http://www.infosharp.org/misp"
  xmlns:gml="http://www.opengis.net/gml">
  <misp:ResponseStatus> ... </misp:ResponseStatus>
  <misp:FeatureCollection>
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>三沢川水門観測所</observatoryID>
        <timestamp>2009-03-11T00:00:00+09:00</timestamp>
        <waterLevel>6.6</waterLevel>
      </RiverInformation>
    </gml:featureMember>
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>保土ヶ谷観測所</observatoryID>
        <timestamp>2009-03-10T12:00:00+09:00</timestamp>
        <waterLevel>5.5</waterLevel>
      </RiverInformation>
    </gml:featureMember>
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>寺家橋観測所</observatoryID>
        <timestamp>2009-03-10T11:00:00+09:00</timestamp>
        <waterLevel>4.4</waterLevel>
      </RiverInformation>
    </gml:featureMember>
    ...
    ...
    ...
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>西ヶ崎橋観測所</observatoryID>
        <timestamp>2009-03-10T13:00:00+09:00</timestamp>
        <waterLevel>1.1</waterLevel>
      </RiverInformation>
    </gml:featureMember>
  </misp:FeatureCollection>
</misp:GetFeatureResponse>

```

図 3 検索結果整列機能の返答例

## ii) 検索結果数およびオフセットの指定

災害情報は時々刻々と蓄積されていくため、最終的には膨大な件数の情報が集積することになる。災害情報の統計的な解析等を行うためには、これらの集積情報をそのまま扱うことになるが、使用頻度としてはそれほど多くないと思われる。一方、比較的利用頻度が高い情報提示

などのためには情報すべてが必要なわけではなく、なにか際立った情報を有限数集め、それを検索・表示するという利用が主体となる。また、多数の検索結果を必要とする場合でも、検索結果を分割して取得する手段を提供する必要がある。このような情報提示の要求に対応するため、検索結果に対して最大フィーチャー数を指定する機能と、検索結果のどの順位から返答として返すかを指定するオフセット機能を MISP に導入し、DaRuMa に実装した。

最大フィーチャー数については、すでに最新版の WFS で定義されているため、それを採用することとした。MISP ではすでに `limit` という属性を Query に指定することができたが、母体とする WFS での仕様に合わせる形で、`maxFeatures` 属性を導入した。また、オフセット機能についてはまだ WFS では定義されていないため、独自拡張を行うこととした。ただし、WFS を制定している OGC の関連規格を調査した結果、地理情報のカタログサービスである CSW (OGC Catalogue Service for Web) <sup>1)</sup> で類似の機能を提供する `startPosition` 属性が規定されていたため、これに準拠することとした。

実際に設計・実装したプロトコルの形式は以下のとおりである。

```
<GetFeature xmlns="http://www.infosharp.org/misp"
  [ startPosition="Offset of features" ]
  [ maxFeatures="Maximum number of features" ]>

  <Query typeName="Name of Toplevel Element"
    <Filter>
      any conditions
    </Filter>

  </Query>
</GetFeature>
```

図 4 検索結果数制限および開始オフセット指定のプロトコル

ここで、各々のパラメータは以下の機能を指定する。

- `startPosition="Offset of features"` :  
返答で結果を返す際のフィーチャーのオフセットを表わす。正の整数でなければならない。この値が N の場合、N 番目のフィーチャーから返す。すなわち先頭から N-1 個のフィーチャーをスキップする。この値のデフォルトは 1 である。
- `maxFeatures="Maximum number of features"` :  
検索で返されるフィーチャーの最大数を表わす。正の整数でなければならない。この属性が省略された場合、検索されたすべてのフィーチャーを結果として返す。`startPosition` と `maxFeatures` を両方指定した場合には、`startPosition` で指定されたオフセットから、最大 `maxFeatures` 個のフィーチャーが返される。

この結果数制限およびオフセット指定を用いた検索例を以下に示す。この例では、すべての水位の観測データに対して水位レベルの上位 2 番目から 4 番目の 3 件の情報を検索するよう指定を行っている。

```

<GetFeature xmlns="http://www.infosharp.org/misp"
             xmlns:rev="http://www.infosharp.org/test/river"
             maxFeatures="3"
             startPosition="2">
  <Query typeName="rev:RiverInformation">
    <Filter> <True/> </Filter>
    <SortBy>
      <SortProperty>
        <PropertyName>rev:waterLevel</PropertyName>
        <SortOrder>DESC</SortOrder>
      </SortProperty>
    </SortBy>
  </Query>
</GetFeature>

```

図 5 検索結果数制限および開始オフセット指定の要求例

```

<misp:GetFeatureResponse ... >
  <misp:ResponseStatus> ... </misp:ResponseStatus>
  <misp:FeatureCollection>
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>保土ヶ谷観測所</observatoryID>
        <timestamp>2009-01-23T07:00:00+09:00</timestamp>
        <waterLevel>8.9</waterLevel>
      </RiverInformation>
    </gml:featureMember>
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>寺家橋観測所</observatoryID>
        <timestamp>2009-02-01T15:00:00+09:00</timestamp>
        <waterLevel>8.5</waterLevel>
      </RiverInformation>
    </gml:featureMember>
    <gml:featureMember>
      <RiverInformation xmlns="http://www.infosharp.org/test/river">
        <observatoryID>多摩川河口観測所</observatoryID>
        <timestamp>2009-02-20T20:00:00+09:00</timestamp>
        <waterLevel>7.9</waterLevel>
      </RiverInformation>
    </gml:featureMember>
  </misp:FeatureCollection>
</misp:GetFeatureResponse>

```

図 6 検索結果数制限および開始オフセット指定の返答例

### iii) グループ内最大・最小値検索機能

集積した共有情報を利活用するためには、ある特定の属性や値が最大や最小を取るもののみを集める機能が必要となる。例えば災害の被害状況が時々刻々報告される場合、最も直前に報告されたもののみを集める必要があるケースが考えられる。

MISP/DaRuMa の設計思想としては、記録された災害情報は削除や上書きされることはなく、更新された情報を新たな情報として蓄積していくという考え方を採用している。これは、データが残されることで、大災害が起こった後の検証においてプロセスの解析が可能となり、その後の災害に対するより良い対応を検討することができることを重視しているためである。すなわち、例えばある地域の被害の状態が変化したことをデータベースに記載する際に、古い状態のデータを削除して新しい状態のデータを登録するのではなく、新たなデータを時刻付きで順次登録するだけにするほうが、時系列での状態の変化を取り出せるため、後の検討を行いやすくする。その反面、状態を削除と登録を用いて上書きしていく方式と比べて、状態の登録のみ

を行う方式では現状の把握が難しいという欠点がある。

現状の把握や最新情報を集約する方法としては、最新時刻の情報のみを集めるという単純な方法が考えられる。これについては、最新時刻、すなわち時刻の値が最大のものを集めればよいので、前節 i), ii) で導入した機能で実現することができる。しかし、これでは単に直前に報告された内容のみが検索されることになり、各特定地域ごとや報告者ごとの最新情報を集める、といったことは難しい。

そこで、i), ii) で導入した整列機能とは別に、指定した情報要素によりグループ化した情報項目集合ごとに最大値・最小値を求める機能を新たに追加することとした。具体的には、検索条件として、全体をある項目が同じである複数の集合に分類し、各集合内で、別のある項目が最大もしくは最小(時刻の場合は最新もしくは最古)のものを全て返す。例えばには地域を表すの項目を分類に使用してグルーピングし、その中で報告時刻が最新(最大)のものを検索することで、各地域における最新情報を集積することができる。

実際に設計した形式は以下のとおりである。

```
<Filter>
  <PropertyIsMax>
    <PropertyName>最大かどうか使用する項目名</PropertyName>
    <UniqueGroupBy>
      <PropertyName>グループ化のための項目名</PropertyName>
    </UniqueGroupBy>
  </PropertyIsMax>
</Filter>
```

図 7 最大値検索の形式

```
<Filter>
  <PropertyIsMin>
    <PropertyName>最小かどうか使用する項目名</PropertyName>
    <UniqueGroupBy>
      <PropertyName>グループ化のための項目名</PropertyName>
    </UniqueGroupBy>
  </PropertyIsMin>
</Filter>
```

図 8 最小値検索の形式

以下は最新の人の状態の集合を取得する例である。人 ID が同じものの中から、報告時刻が最新のものを全て返す。人 ID が複数あれば、それらは別々に扱われそれぞれで最新のものを返す。すなわち現在保持している最新の状態を取得することになる。

```
<Filter>
  <PropertyIsMax>
    <PropertyName>u:reportDate</PropertyName>
    <UniqueGroupBy>
      <PropertyName>u:personID</PropertyName>
    </UniqueGroupBy>
  </PropertyIsMax>
</Filter>
```

図 9 検索結果整列機能の要求例

#### b). 汎用性・可搬性の拡張

本プロジェクトで開発を進めている情報共有システムの枠組みは、さまざまな既存システムと連携することを前提としている。このため、データベースなどシステムが稼働させることのできる計算機環境などはできるだけ広範囲であることが望まれる。そこで本課題では開発を進めているデータベースシステム DaRuMa の汎用性・可搬性を高めるため、サポートする OS やバックエンドデータベースを多様化することを行った。

#### i). PostGIS 対応

従来の DaRuMa は、バックエンドのデータベースとして、オープンソースのデータベースシステムとして最も広く用いられている MySQL を用いていた。MySQL はその高速性と柔軟性に加え、地理データを直接サポートしていることから地理情報の扱いが必須となる災害情報のためのバックエンドデータベースとしては適していた。一方、トランザクション機能の欠如など確実性を重視する用途では敬遠されるといった問題点もあった。

一方、日本においては同じオープンソースのデータベースシステムである PostgreSQL も広く用いられており、MySQL と同じく幅広いプラットフォームをサポートしている。また、最近では地理情報を扱うライブラリを組み込んだ PostGIS も開発されており、MySQL と並んで災害情報のバックエンドとして有力な候補となる。特にトランザクション処理機能を持っており、エンタープライズ用途としても利用できる品質を持っている。

そこで本プロジェクトでは DaRuMa のバックエンドとして PostGIS もサポートし、利用者が各々の計算機環境や用途によりバックエンドを選択できるよう、改良を進めた。今回の改良により、Java で記述されたミドルウェアをまったく変更することなく、起動時の設定ファイルを変更することでバックエンドのデータベースを切り替えることを可能とした。



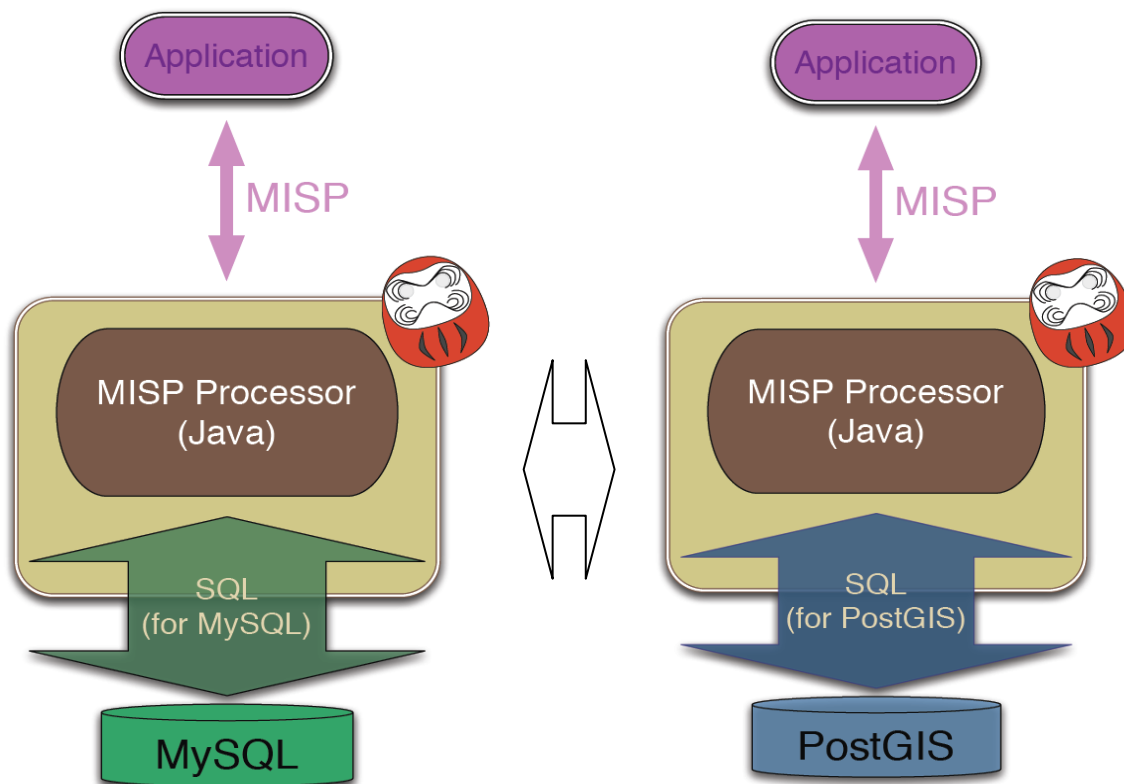


図 10 PostGIS 対応

また今回の改良において、性能的な向上も確認できた。大量のデータを扱うときは、MySQL を使用した場合に比べて、データ登録時(Transaction/Insert)で 5~15%ほど、データ取得時(GetFeature)で 3~5%ほど高速に動作することが確認できた。

バックエンドのデータベースシステムを 2 種類以上もつよう改良することには、さらなるメリットがある。一般に複数のデータベースに対応するためには、それらのプロトコルの共通機能のみを利用するように設計する必要があるが、これによりデータベースとの接続が整理され、より標準に近い機能のみを用いてデータのやり取りを行うことになる。実際、今回の改良により PostGIS だけでなく、市販のデータベースである Oracle や Microsoft SQL Server などへの対応も可能な形での再設計・実装を行うことができた。これらのデータベースの実際のサポートは今後の課題であるが、利用機能の整理による内部構造の簡素化は、今後の普及や改良において重要なファクターとなると期待できる。

#### ii) バックエンドデータベース切り替え機能

本プロジェクトでは災害時の情報システム間の情報共有基盤を提供することを目標としているが、災害専用システムではなく、平常時も利用できるシステムとして設計している。これは、平常時から利用していないシステムは非常時にも利用できないという考えによる。この場合、平常時に使用しているデータをそのまま非常時に利用することは、プライバシーやセキュリティの面から難しい。そこで、平常時と非常時をスムーズに切り替える機能を提供するため、バ

ックエンドのデータベースを複数もち、それを用意に変更できる構成を取れるよう、DaRuMa の改良を行った。

機能的には前節で述べた PosGIS 対応と同じく、設定ファイルにより使用するバックエンドデータベースを選択する構成をとった。これにより、DaRuMa の利用局面でデータベースを切り替える他、一台の計算機上に複数の DaRuMa を起動し、それらが別々のデータベースをバックエンドとして利用することが可能になる。

この機能により、DaRuMa の活用範囲を広げることができ、普及や改良を容易にできると考えられる。

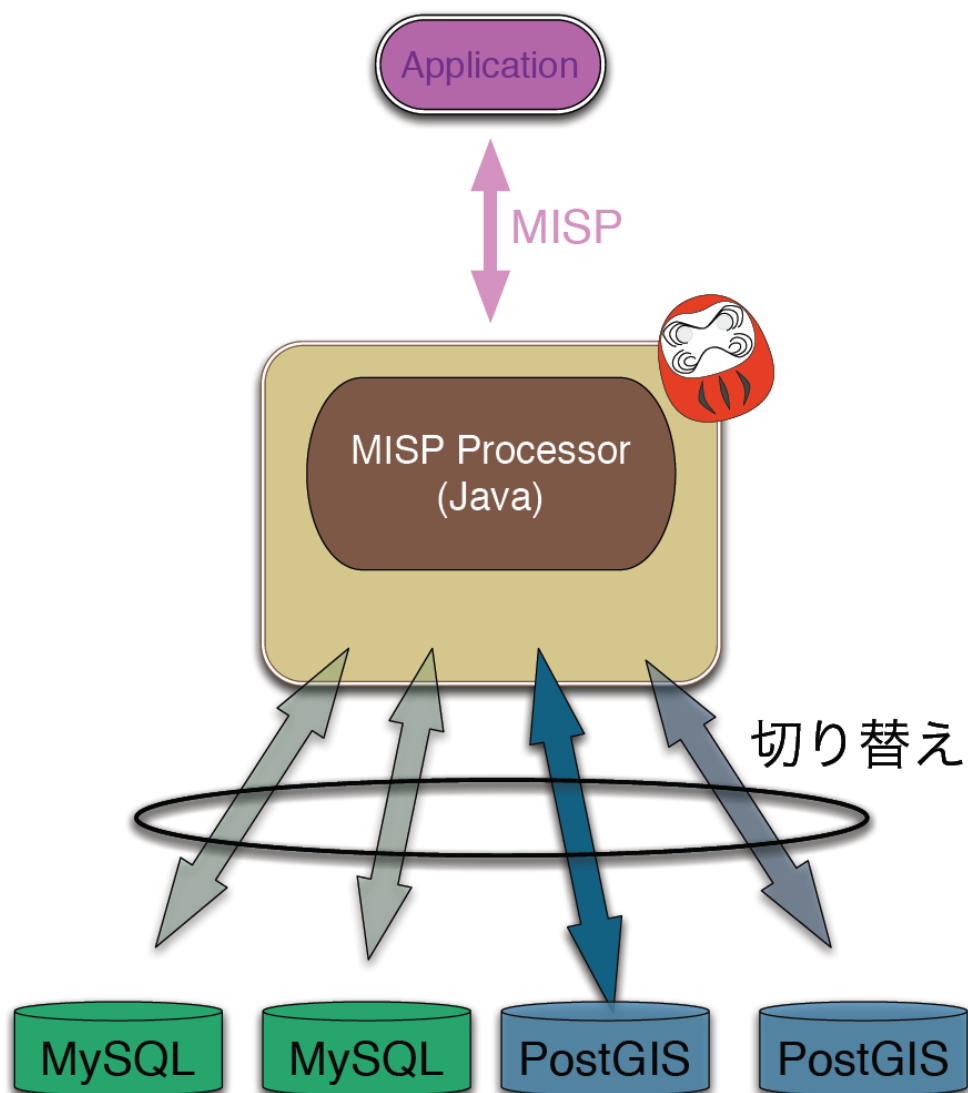


図 11 バックエンドデータベースの切り替え機能

### iii) 対応 OS 強化

可搬性を高めるもうひとつの拡張として、対応 OS の強化を進めた。DaRuMa はこれまで

Windows, Linux, FreeBSD をサポートしてきたが、これに MacOS を加え、一般に使われている主な OS をほぼカバーできるように改良を行った。(図 12)

MacOS はデスクトップを中心にシェアを伸ばしている OS であり、主としてサーバで用いられることの多い Linux や FreeBSD に比べ普及台数も多い。この上で動作が可能となることで、DaRuMa の動作環境依存性を削減でき、普及および長期にわたり活用を目指すことができる。

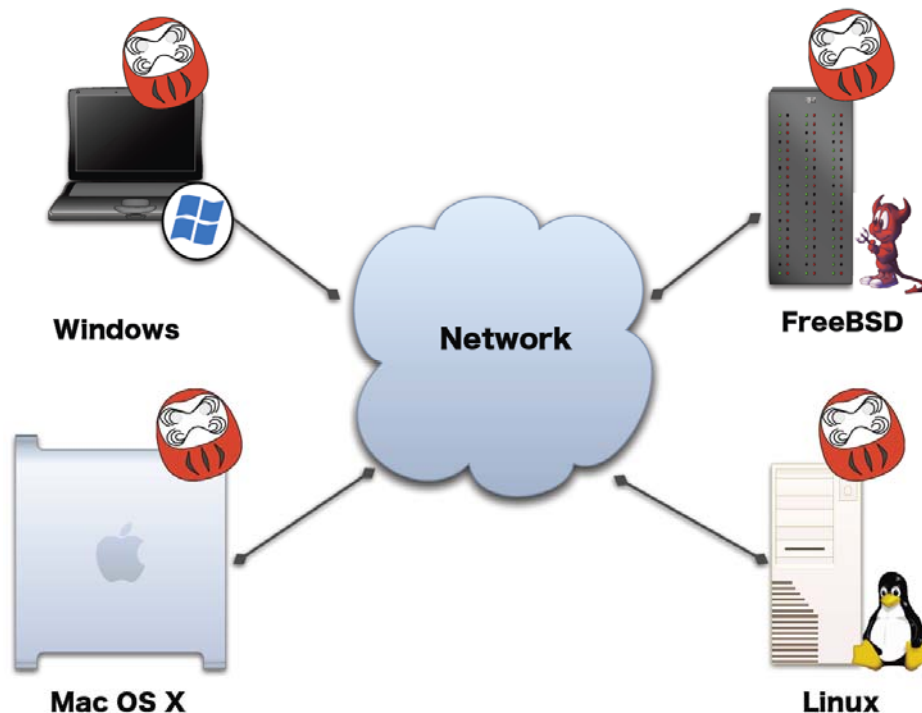


図 12 複数 OS 対応

#### iv) USB DaRuMa

災害時には情報インフラも被害を受ける可能性がある。よって災害情報システムを構成する際には、どのようにバックアップシステムを用意するかが大きな課題となる。一般にバックアップシステムは、主となるシステムの完全なコピーを別途用意しておく方式をとることが多いが、コストや維持作業も 2 倍になるなど問題点も多い。これを解決するひとつの方策として、いざというときに任意の計算機で DaRuMa を稼働できる手段を提供する方法を考案した。

われわれはここ 5 年に発売された多くの PC が USB から起動できる機能を持っていることに着目し、USB に DaRuMa を含む OS を格納し、そこからネットワーク機能、サーバ機能、DaRuMa 機能を自動的に立ち上げるよう設定を行い、煩雑な設定作業なしに DaRuMa 機能を提供するシステムを構成した。この USB 起動型システムを USB DaRuMa (日本名ゆびダルマ) と呼ぶ。

USB DaRuMa は以下の構成をとっている。

- Ubuntu Linux
- Avahi および zero conf による自動的ネットワーク設定機能
- Apache による Web Server 機能および MISP on HTTP 機能。

- USB DaRuMa 複製機能

本システムの特徴は、必要とする機材が 2Gbyte 以上の USB メモリのみであるところで、これにより任意のあいている計算機を DaRuMa サーバとして活用できる点である。また利用方法も簡便であり、図 13 に示したように、USB DaRuMa を格納した USB メモリを差し込んで PC を起動するだけで DaRuMa サーバを立ち上げることができる。



図 13 USB DaRuMa とその起動の様子

この USB DaRuMa によるシステムは、以下のような運用を想定している。通常のクライアントサーバシステムにおいては、インフラの被害によりサーバへのアクセスができなくなった場合、まったく機能しなくなってしまう。このような事態に対して USB DaRuMa で立ち上げた最小構成の DaRuMa をサーバとして立ち上げることができれば、利用できる情報に制約がありつつも、庁舎内や一自治体内で小規模に、情報入力や部分的な情報共有機能を提供することが可能となる。インフラが復旧するに従い、USB DaRuMa に格納された情報を本格運用される中核の DaRuMa サーバに集約していくことで、首都直下全域での情報共有に円滑につなげていく。

また、この USB DaRuMa には DaRuMa 以外のツールを搭載することも可能であり、ビューワや情報提供用のシステムなどを同胞することで、サーバだけでなく情報共有のための端末としても利用できると思われる。

## 2) 連携の有効性検証と拡張

本プロジェクトでは、既存の各種システムを MISP/DaRuMa を通じて連携させ、災害救助に関わる情報共有を実現することを目指している。これに向けて、本年度は道路情報のスキーマの整理と、救急医療情報の情報共有における MISP/DaRuMa の有効性の検証、さらには地理情報・センサー情報などの標準化との連携活動を進めた。

### a) 道路情報スキーマ

道路状況の情報は、広域における災害情報共有で最も重要性が高いものになる。このため本年度は道路情報の標準的な情報項目を整理し、それに基づき XML の表現形式を設計した。

図 14 は整理を行った情報項目（一部抜粋）であり、図 15～図 16 はそれら表現する XML 表現を XML Schema により記述したものである。

項目	プロパティ名	データ型	min	max
道路	RoadType			
道路 ID	roadId	xsd:string	1	1
路線名	roadName	xsd:string	1	1
路線番号	roadNumber	xsd:integer	1	1
橋梁名	bridgeName	xsd:string	0	1
道路区分	roadClassification	xsd:string (国道   首都高速道路   都道・県道   その他一般道)	1	1
道路区間	roadPath	RoadPathType	1	1
道路被害 ID	roadDamageId	xsd:string	0	1
道路規制 ID	roadRegulationId	xsd:string	0	1
道路混雑 ID	roadTrafficId	xsd:string	0	1
緊急交通路 指定想定路線 ID	candidateEmergencyRoadId	xsd:string	0	1
緊急交通路 ID	emergencyRoadId	xsd:string	0	1
緊急輸送路 ID	emergencyTransportRouteId	xsd:string	0	1
備考	note	noteType	0	1
道路区間	RoadPathType			
リンク ID	linkId	xsd:string	1	1
始点ノード	startNode	xsd:string	1	1
終点ノード	endNode	xsd:string	1	1
道路形状	roadGeometry	gml:GeometryPropertyType(LineString)	1	1
距離	roadLength	xsd:float	1	1
幅員区分	roadWidthClassification	xsd:integer (1: 13m 以上   2: 5.5 以上 13.0 未満   3: 3.0 以上 5.5 未満   4: 3.0 未満   0: 不明)	1	1
総車線数	roadNumLanes	xsd:integer	0	1
道路被害	RoadDamageType			
路線名	roadName	xsd:string	1	1
路線番号	roadNumber	xsd:integer	1	1
橋梁名	bridgeName	xsd:string	0	1
被害箇所住所	damagePointAddress	xsd:string	1	1
道路区間	roadPath	RoadPathType	1	1
道路 ID	roadId	xsd:string	1	1
道路被害 ID	roadDamageId	xsd:string	1	1
	damageConfirmedTime	gml:TimePositionType	1	1
被害区分	damageClassification	xsd:string (陥没   隆起   土砂災害   崩落   冠水   建物崩壊   ブロック塀倒壊   電柱倒壊   自動車による閉鎖等   落橋   橋脚橋台破損   橋桁破損   越水   その他)	1	1
備考	note	noteType	0	1

図 14 道路データの情報項目とその属性 (一部)

```

<xsd:schema
  xmlns="http://www.infosharp.org/schemas/dml"
  xmlns:misp="http://www.infosharp.org/misp"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:dml="http://www.infosharp.org/schemas/dml"
  misp:id="http://www.infosharp.org/schemas/dml/road.xsd"
  targetNamespace="http://www.infosharp.org/schemas/dml">

  <xsd:import
    namespace="http://www.opengis.net/gml"
    schemaLocation="http://schemas.opengis.net/gml/3.1.1/base/feature.xsd"/>
  <xsd:import
    namespace="http://www.infosharp.org/schemas/dml"
    schemaLocation="http://www.infosharp.org/schemas/dml/message.xsd"/>
  <xsd:import
    namespace="http://www.infosharp.org/schemas/dml"
    schemaLocation="http://www.infosharp.org/schemas/dml/roadpath.xsd"/>
  <xsd:import
    namespace="http://www.infosharp.org/schemas/dml"
    schemaLocation="http://www.infosharp.org/schemas/dml/note.xsd"/>

  <xsd:element name="Road" type="RoadType" substitutionGroup="gml:_Feature"/>

  <xsd:complexType name="RoadType">
    <xsd:sequence>
      <xsd:element name="roadId" type="xsd:string"
        minOccurs="1" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadName" type="xsd:string"
        minOccurs="1" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadNumber" type="xsd:integer"
        minOccurs="1" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="bridgeName" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadClassification" type="xsd:string"
        minOccurs="1" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadPath" type="dml:RoadPathType"
        minOccurs="1" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadDamageId" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadRegulationId" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="roadTrafficId" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="candidateEmergencyRoadId" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="emergencyRoadId" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="emergencyTransportRouteId" type="xsd:string"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
      <xsd:element name="note" type="dml:noteType"
        minOccurs="0" maxOccurs="1" misp:indexed="false"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>

```

図 15 道路情報(RoadType)のスキーマ

```

<xsd:schema
  xmlns="http://www.infosharp.org/schemas/dml"
  xmlns:dml="http://www.infosharp.org/schemas/dml"
  xmlns:misp="http://www.infosharp.org/misp"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  misp:id="http://www.infosharp.org/schemas/dml/roadpath.xsd"
  targetNamespace="http://www.infosharp.org/schemas/dml">

  <xsd:import
    namespace="http://www.opengis.net/gml"
    schemaLocation="http://schemas.opengis.net/gml/3.1.1/feature.xsd"/>

  <xsd:element name="RoadPath" type="RoadPathType" substitutionGroup="gml:_Feature"/>

  <xsd:complexType name="RoadPathType">
    <xsd:sequence>
      <xsd:element name="linkId" type="xsd:string" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="startNode" type="xsd:string" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="endNode" type="xsd:string" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="startPointName" type="xsd:string"
        minOccurs="0" maxOccurs="1"/>
      <xsd:element name="endPointName" type="xsd:string"
        minOccurs="0" maxOccurs="1"/>
      <xsd:element name="roadGeometry" type="gml:GeometryPropertyType"
        minOccurs="1" maxOccurs="1"/>
      <xsd:element name="roadLength" type="xsd:float" minOccurs="1" maxOccurs="1"/>
      <xsd:element name="roadWidthClassification" type="xsd:string"
        minOccurs="1" maxOccurs="1"/>
      <xsd:element name="roadNumLanes" type="xsd:integer"
        minOccurs="0" maxOccurs="1"/>
      <xsd:element name="note" type="dml:noteType" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>

```

図 16 道路区間情報のスキーマ

また、図 17 は上記のフォーマットで表された南関東の主要道路の情報を画面に表示したものである。実際の利用場面ではこれらの道路基盤情報に被災状況や渋滞状況、プローブカーによる現況情報が重畳されることになるが、基盤形式が定められたために、今後は情報付加が容易になると考えられる。



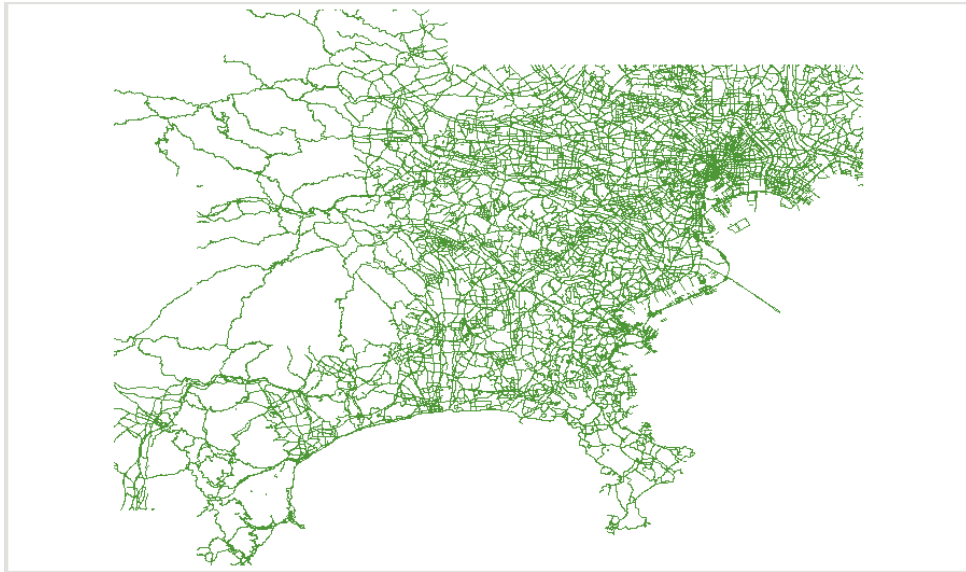


図 17 道路情報の表示例

#### b) 広域災害救急医療情報システムとの連携の有効性検証

広域災害救急医療情報システムとの連携について、順天堂大学附属浦安病院救急診療科に協力し、連携による情報統合および情報共有の有効性の検証を進めた。

現在、大災害時における救助活動の効率化を高めるために、県単位および県をまたがった範囲での救急医療の被災・受容情報を集約するものとして広域災害救急医療情報システム(EMIS)が運用されているが、このシステムではテキストによる情報提示が主となっており、また、医療情報のみに特化して情報収集・提示を行う形式をとっている。よって、広域における医療機関同士の位置関係や道路被害状況など他の被災情報との関係を把握することが困難であると考えられる。一方、我々が開発を進めている情報共有のプラットフォームでは医療情報・道路情報・停電情報などを集約・管理することが可能となっている。

この EMIS を含めた情報共有については、昨年度より順天堂大学岡本氏らと共同でシステム連携を進めてきたが、今年度はこの集約・情報共有の有効性を示すため、情報共有プラットフォームによる情報統合が重症患者の転送先施設決定作業にどの程度効果をもたらすかを検証する実験を試みた。

実際に行った実験の設定を図 18 に示す。この図のように、EMIS のテキストベースの情報を中心とした現状の意思決定作業と、DaRuMa を介して GoogleEarth 上に統合表示した情報をもとに行った意思決定作業を複数の被験者に行ってもらい、作業時間の計測や、使い勝手についてのアンケート調査を行った。図 19 はその結果である。この結果らわかるように、情報統合の効果は高く、既存のシステムを DaRuMa により統合するだけでも有効なシステム構成が可能であると考えられることができる。

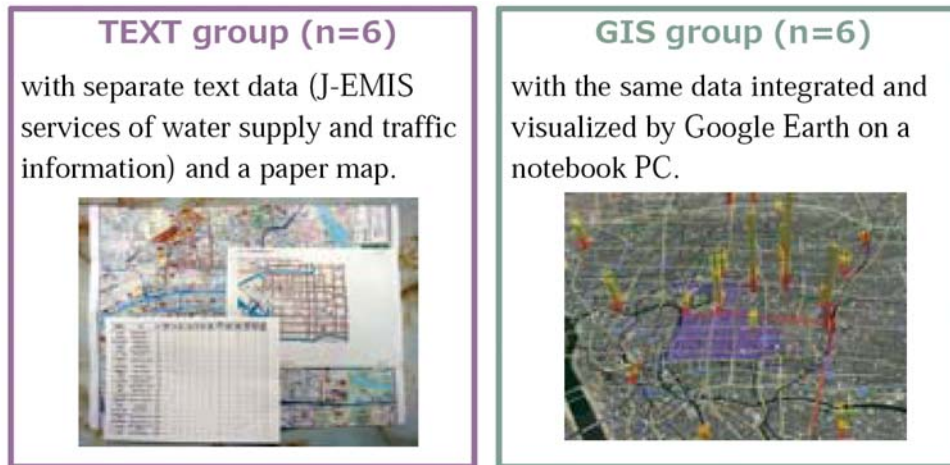


図 18 EMIS と道路情報の分離・統合表示

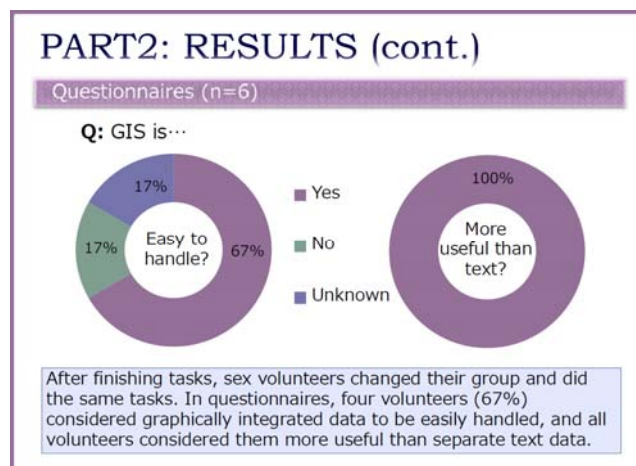
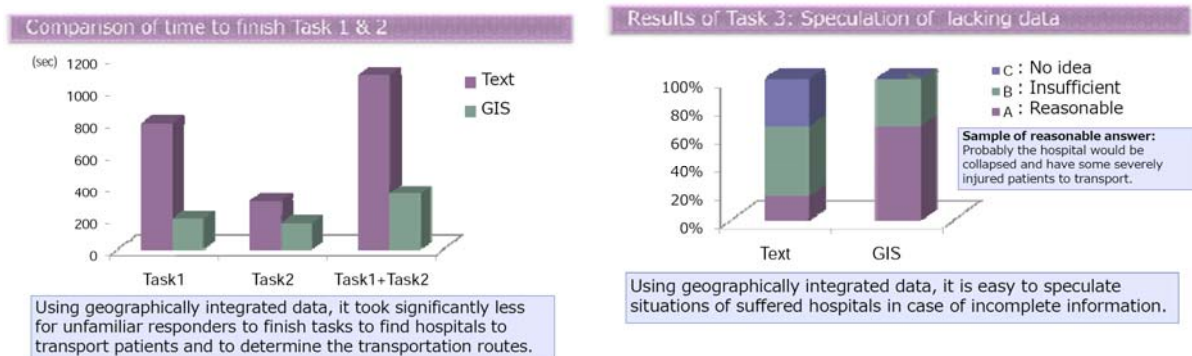


図 19 DaRuMa による情報統合の効果

(c) 結論ならびに今後の課題

本サブテーマでは、当初の予定通り、来年度以降に予定されている評価実験のシナリオに沿って他グループで実装が進められている各システムの連携に必要な機能の拡張項目を洗い出し、特に検索機能の充実と汎用性・可搬性の確保を中心に MISP の再設計および DaRuMa の改良を

進めた。機能の充実については特に東京大学の竹内グループで進められている汎用ビューワの開発と連携し、拡張機能の細かい仕様まで意見交換を行いながら調整を行ってきた。これにより、DaRuMa/MISP をより実用的なものとする事ができたと考えている。

また、評価実験に向けて具体的なコンテンツを取り上げ、そのスキーマの設計やシステム連携の効果の検証も進める事ができた。

ただ、これらの改良・開発は主としてボトムアップになされたため、システムのスケラビリティや標準化に向けてトップダウンに整理する必要が出てきていることも確かである。来年度はこの整理を行いつつ、評価実験によるシステム実装を進めていく必要がある。

(d) 引用文献

- 1) OGC: “OpenGIS Web Feature Service (WFS) Implementation Specification”,  
<http://www.opengeospatial.org/standards/wfs> .
- 2) OGC: “OpenGIS Filter Encoding Implementation Specification”,  
<http://www.opengeospatial.org/standards/filter> .
- 3) OGC: “OpenGIS Catalogue Service Implementation Specification”,  
<http://www.opengeospatial.org/standards/cat> .

(e) 学会等発表実績

学会等における口頭・ポスター発表

発表成果(発表題目、口頭・ポスター発表の別)	発表者氏名	発表場所 (学会等名)	発表時期	国際・国内の別
道路交通での速度違反取り締まりにおける確率的な罰政策の効果(口頭)	松井宏樹, 野田五十樹, 和泉潔	人工知能学会全国大会	2008.06.13	国内
Integrated Simulation and Information Sharing System for Disaster Mitigation (招待講演)	Itsuki Noda	8th International Conference on Cellular Automata for Research and Industry (ACRI 2008)	2008.09.23	国際
Efficiency of Integration of Emergency Management Information Systems By a Geographical Feature Database Service(ポスター発表)	Ken Okamoto, Itsuki Noda, Tomoyuki Yoshiwara, Yasumasa Ode, Shigeru Matsuda, Yoshiyasu Yamada, Hiroshi Tanaka	American Association for the Surgery of Trauma, the 67th meeting	2008.09.27	国際

学会誌・雑誌等における論文掲載

掲載論文（論文題目）	発表者氏名	発表場所 （雑誌等名）	発表時期	国際・国内の別
IT Framework for Disaster Mitigation Information Sharing	Itsuki Noda, Hiroki Shimora, Hiroki Matsui, Hiroshi Yokota, Akihiro Shibayama, Yoshiaki Hisada, Masahiro Murakami, Takeyasu Suzuki, Yasunori Hada, Takeshi Yamada, Shinsaku Zama, Yasushi Hada, Jun-ichi Meguro, Ken Okamoto	Journal of Disaster Research	2008.12.31	国際

マスコミ等における報道・掲載

(なし)

(f) 特許出願, ソフトウェア開発, 仕様・標準等の策定

1) 特許出願

なし

2) ソフトウェア開発

名称	機能
減災情報共有データベース (DaRuMa) ver. 20090331	Web Service 化、汎用性・可搬性強化、検索機能強化

3) 仕様・標準等の策定

なし

(3) 平成21年度業務計画案

前年度までに拡張したデータベースプロトタイプシステム (DaRuMa) に、広域連携の評価実験 (システム連携のデモ) のためのデータを登録し、実験に参画する。そのために、個別研究テーマ (2) の他グループで開発するシステムとの連携ツールの開発および改良を進める。さらに、評価実験から抽出された課題に基づいて、DaRuMa/MISP の開発項目を精査し、次年度以降の開発計画を立案する。