

Web GIS アプリケーションフレームワーク rinzo.ma の設計と応用

横山 昌平^{†1} 黒井 星 良^{†2}
岡本 章 裕^{†3} 石川 博^{†1}

本論文では Web GIS アプリケーションフレームワーク rinzo.ma の設計と応用について紹介する。rinzo.ma は衛星画像を用いてリモートセンシングを行う研究者のために設計された基盤ソフトウェアであり、Web ブラウザ上で動作する。rinzo.ma は柔軟なプラグイン仕様を持っており、また公開、共有、再利用が容易である。Web GIS アプリケーション開発者は、公開されているプラグインや自身で実装したプラグインを組み合わせ、様々な空間データの検索や処理を効率的に実現することができる。本論文では実証例として、rinzo.ma の応用、たとえば地震前後の ASTER 衛星画像検索閲覧システム等、現在実際に稼働している複数のアプリケーションを紹介する。

Design and Application of a Web GIS Application Framework, rinzo.ma

SHOHEI YOKOYAMA,^{†1} SEIRA KUROI,^{†2}
AKIHIRO OKAMOTO^{†3} and HIROSHI ISHIKAWA^{†1}

In this paper, we describe design and application of a Web GIS application framework, rinzo.ma. The rinzo.ma is a infrastructure software which is available on most Web browsers, and it is designed for researchers of remote sensing using satellite images. Using rinzo.ma developers can install easily plug-in software which includes geospatial data and process into the own GIS application. We also describe some applications of rinzo.ma, includes the satellite image retrieve and browse system for the before and after earthquake.

1. はじめに

近年、Web 技術の急速な発展にともない、たとえば Google Maps に代表されるように、衛星画像等 GIS が扱う時空間データ（以下 GIS データと呼ぶ）を Web ブラウザで簡単に閲覧する仕組みに注目が集まっている。Web2.0 というキーワードとともに注目を浴びた JavaScript/Ajax やダイナミック HTML を駆使すれば、地図のスクロールや動的なデータ読み込みが Web ブラウザ上で可能となった。このように Web コンテンツがリッチなユーザインタフェースを持つことにより、従来ならばスタンドアロン型として提供されていた種々のアプリケーションが Web ブラウザ上で動作する軽量のサービスとして、次々に無料あるいは安価に提供されている。

しかしながら、そのような Web 技術の急速な発展の恩恵は、同じく衛星画像を使う地球科学・リモートセンシング等の研究用にまでは波及していない。本論文で提案する Web GIS アプリケーションフレームワーク rinzo.ma は衛星画像を用いる研究者のための基盤ソフトウェアであり、Google Maps のようなリッチインタフェースを持った GIS アプリケーションを Web ブラウザ上で構築するためのフレームワークである。rinzo.ma を利用することにより、ユーザ独自の衛星画像検索、処理、閲覧システムを Web ブラウザ上に rinzo.ma のプラグインとして構築することができる。

rinzo.ma プラグインは公開・共有することができる。また公開されている複数のプラグインを組み合わせ、独自の Web GIS アプリケーションを構築することも可能である。rinzo.ma がプラグイン作成者に提供するものは、統一的な GUI 設計、プラグインの共通仕様、およびプラグイン間でのデータ交換の仕組みである。本論文では、その共通仕様について詳述するとともに、その設計へ至る背景を説明する。rinzo.ma は産業技術総合研究所による地球観測衛星データの大規模アーカイブ・高度処理に関する研究プロジェクトである GEO Grid との連携のもとで設計され、オンデマンド DEM (Digital Elevation Map/数値標高モデル) マップ作成のためのインタフェースとして実際に利用されている。このシステムでは rinzo.ma フレームワーク上で、Terra 衛星搭載 ASTER⁸⁾ が観測した 160 万枚を超える

^{†1} 静岡大学情報学部
Faculty of Informatics, Shizuoka University

^{†2} スズキ教育ソフト
Suzuki Educational Software

^{†3} 静岡大学大学院情報学研究科
Graduate School of Informatics, Shizuoka University

衛星画像から、容易に目的の場所・撮影日時 of 画像を検索する仕組みを実現した。このほかにも、Terra, Aqua 衛星搭載 MODIS⁵⁾ センサによる黄砂観察や、地震前後の震源地付近の衛星画像を容易に検索・閲覧するシステムを作成している。rinzo.ma が扱うのは衛星画像だけではない。地質図の表示や緯度経度が整数値で交わる場所のデジタルカメラ写真を公開している Degree Confluence Project のビューワや、住所を緯度経度に変換するジオコーディング機能等も rinzo.ma プラグインとして公開している。

本論文では、rinzo.ma の設計をデータベース・JavaScript 実装の視点から詳述する。また前述した実際に使われているプラグインをユースケースとし、rinzo.ma の拡張性や利便性等を紹介する。

続く 2 章では関連する研究について述べる。3 章では rinzo.ma フレームワークについて詳述し、4 章では rinzo.ma プラグインの公開・共有・連携手法について述べる。さらに 5 章で実際に利用されているプラグインの設計について述べる。6 章で本論文をまとめ、今後の課題を考察する。

2. 関連研究

GIS データを Web 技術を使って検索、閲覧する手法はすでに Open Geospatial Consortium (OGC)⁴⁾ において標準化が進められており、画像データを配信する Web Map Service (WMS)、地物データを検索する Web Feature Service (WFS)、カバレッジデータを検索する Web Coverage Service (WCS)、メタデータ検索のための Catalogue Service for Web (CS-W) が提案されている。しかしながら、これらはデータの配信や検索のための仕組みであり、インターネット上に分散して存在する様々な GIS データを統合する手法ではない。rinzo.ma は柔軟なプラグイン仕様を持っており、これら OGC 標準の GIS データを、Web ブラウザ上で統合利用 (マッシュアップ) することができる。

このような GIS データ統合利用は OWS-4³⁾ において BPEL を使ったワークフローが提案されている。しかしながら、この BPEL 等のワークフロー方式では、サービスを直列に接続し手続的に実行することを基本としており、本研究が目指す分散しているデータの処理や検索を有機的に連携利用する Web GIS アプリケーションを実現することはできない。提案手法 rinzo.ma はプラグインとしてデータ検索、閲覧、処理機能を Web アプリケーションに動的に加えていく方式をとっている。これにより、複数の異なるデータ・処理手法を連携させ、かつそのアプリケーションを統一的な GUI 画面で操作することができる。

GEON⁶⁾ はグリッド技術に基づいた地理情報基盤技術でありオントロジを考慮したデータ

検索を行うことができる。また e-science の他分野においてはたとえば International Virtual Observatory Alliance (IVOA)¹⁾ がデータ統合を SQL に準ずる VOQL²⁾ という問合せ言語を用いて行う手法を提案している。これらの手法はデータ検索のための手法であり、不均質な複数の GIS データの統合的な利用を行うことはできない。rinzo.ma では、単に検索やデータ統合にとどまらず、GUI までも含めた GIS サービスの統合を目指す。

異種の GIS データ統合を Grid 技術で解決を目指す研究として産業技術総合研究所の GEO Grid⁷⁾ がある。グリッド環境上に複数の GIS データソースを含む Virtual Organisation と呼ぶ仮想的なネットワークを作ることで、データのセキュリティを担保しつつ柔軟なデータ統合、処理ワークフローを実現している。rinzo.ma は GEO Grid と協力の下で、そのいくつかのアプリケーションでユーザインタフェースとして採用されている。

3. rinzo.ma フレームワーク

本章ではまず rinzo.ma 開発に至る背景を述べる。rinzo.ma は産業技術総合研究所 GEO Grid プロジェクトの一環として開発を開始し、その設計にはリモートセンシング等の研究者のニーズを取り入れている。章の後半では、rinzo.ma フレームワークを詳述する。

3.1 背景

GEO Grid プロジェクトでは、Terra 衛星搭載 ASTER センサが取得したデータ (以下 ASTER データと呼ぶ) を全量グリッド環境で保存している。そのデータを使い、たとえば、オンデマンドで DEM マップを生成したり、植生の変化を調べたりという用途で利用されている。ASTER データに限ったことではないが、センシングデータのデータ量は膨大であり、また日々の計測によりデータ量は漸増する。そのような膨大なデータの中から、目的のデータを見つけることは困難である。もちろん衛星画像には、メタデータが付いており、画像の四隅の位置情報 (緯度経度) や撮影時刻等の情報は保持している。それらを利用すれば日時・場所を指定した検索を実現することはできるが、メタデータに表れない項目については検索ができない。メタデータに表れない項目とは、衛星画像の例でいえば、画像中の観察したい地点に雲がかかっているかどうかということである。このように地球観測データには、人の目で見て判断するということが非常に重要である。

メタデータは ISO19115 等の標準技術がすでにあり、網羅的な約 300 個の項目が定義されている。また ISO19115 等のメタデータを検索するカタログサービスは OGC CSW として標準化されている。しかしながら、特定のアプリケーションの実装を考えると、これら網羅的な検索システムをユーザが直接使うことは考えにくい。なぜならば、データソース

が持つメタデータは限られており、またアプリケーションから利用する項目は、その中のさらに限られた項目だけである。CSW は検索のための独自の問合せ言語を持っており、関係データベースに対する SQL のように、ユーザがカタログサービスに対して任意の問合せを発行することができる。しかし、リモートセンシング研究者等のエンドユーザに、この問合せ言語、ISO19115 によるスキーマ、そして目的のデータソースが利用しているメタデータ項目に関する知識を要求することは困難である。エンドユーザが利用するのはあくまでもこれら標準化技術に基づいたアプリケーションであり、GUI (Graphical User Interface) による操作である。

つまり地球科学研究に向けた GIS アプリケーションでは、ユーザが見て判断することが重要であり、またデータ検索も単にデータベースと問合せ手法だけ用意するのではなく、GUI の構築を含めて考える必要がある。前述した CSW にはすでにいくつかのクライアントシステムが存在する。たとえば deegree は特定のアプリケーションによらない Java による CSW クライアントを提供している。これを使い、公開された CSW サービスに対して、クライアントアプリケーションを実装することができる。しかしながら、クライアントごとの連携は考えられていないため、ユーザごとあるいはアプリケーションごとに、それが利用する個々の CSW サービスすべてに対し問合せや GUI そして連携の仕組みを設計・実装しなければならない。つまり、ある CSW サービスに対しクライアントを実装したからといって、それを再利用する仕組みが現状ではない。本論文では、この再利用を容易にする仕組みとして、我々が開発した Web GIS アプリケーションフレームワーク rinzo.ma の設計と応用を報告する。

rinzo.ma は単に GIS データを使ったアプリケーションの実装をサポートするフレームワークにとどまらず、GIS アプリケーションの再利用や、GIS アプリケーションどうしの連携、複数の GIS アプリケーションによるワークフロー等を実現する。

3.2 設計

3.2.1 設計要件

前述した CSW を利用した GIS アプリケーションでは、データを有する人(データ提供者)ではなく、GIS アプリケーションを実装するデータの利用者が問合せや GUI を含めたクライアントシステムを構築しなければならない。これは関係データベース等を利用したシステムでは一般的な方法である。たとえば社員録のデータベースを考えると、それを利用するアプリケーションごとに、必要な問合せや、それをグラフィカルに操作するための GUI を開発する必要がある。しかしながら、GIS データは多種多様であり、それぞれが特化さ

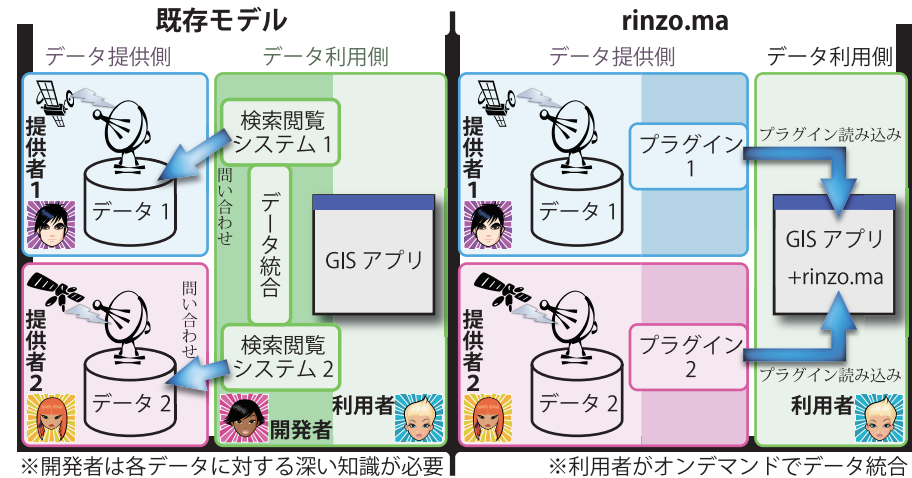


図 1 rinzo.ma のソフトウェアモデル
Fig. 1 Software model of rinzo.ma.

れたメタデータスキーマを持つ。ISO19115 により網羅的なメタデータ仕様はあるが、それら膨大なメタデータ項目のうち、ある GIS データが持つメタデータは数個であるかもしれない。それら複数の GIS データを保持する CSW サービス(データベース)は、非常に疎でかつ異種のデータを単一の網羅的なスキーマで扱うことになる。各 GIS データが利用するメタデータ項目はデータソースごとに異なり、自明ではないので、利用者は実際に GIS データの検索を行う前に、まずはどのメタデータ項目が有効なのかについての知識が必要となる。地球環境科学の研究者が利用者となる GIS アプリケーションにおいて、膨大な数の GIS データに対するメタデータスキーマの知識を前提とすることは困難である。

本論文で詳述する rinzo.ma がこれら既存手法と異なるのは、GIS データ提供者が検索や利用のためのユーザインタフェースまで作成するというモデル(図 1)を採用している点である。GIS データの利用には、メタデータスキーマの知識が必要と前述したが、その知識を深く有しているのはデータ提供者である。つまりデータ提供者がメタデータ検索用の GUI を作成することにより、データの利用者は、メタデータに関する知識がなくても GUI 操作により、必要な GIS データを検索し利用することができる。言い換えるなら、従来はシステム開発の技術を持ち、また利用するデータに対する知識を持っている開発者が利用者のニーズに応じたシステムを個々に開発していたのに対し、rinzo.ma では、データ提供者が

rinzo.ma プラグインの形式として、データ検索・処理を操作するユーザインタフェース含んだコンポーネントとして GIS データが提供されるため、利用者は rinzo.ma 上でそれらプラグインを組み込むだけで複数の GIS データを連携利用できる。

rinzo.ma は非常に柔軟なプラグインの仕様を持っている。データ提供者はデータ検索のためのシステムと GUI を rinzo.ma のプラグインとして実装することにより、データの利用者は Web ブラウザのみでこれを利用できる。またプラグインどうしの連携をサポートする仕組みも有しており、複数のデータソースの連携も可能である。たとえば住所から緯度経度を算出するジオコーディングのプラグインと衛星画像検索プラグインを連携させ、住所からその場所の衛星画像を検索する等の利用が考えられる。

rinzo.ma を利用する際に必要なソフトウェアは Web ブラウザだけである。プラグインの動的な追加や組み替えも Web ブラウザだけで行うことができる。特定のアプリケーションのインストールはいっさい必要ない。GEO Grid プロジェクトでは ASTER センサのデータを全量オンデマンドで有しており、観測の数日後には画像として利用することができるようになる。つまり僻地においてもインターネット環境と Web ブラウザさえあれば、その最新の衛星画像を含むすべての画像にアクセスすることが可能となる。

これまで述べてきた設計要件を以下にまとめる。

- Web ブラウザのみで動作
- RIA (Rich Internet Application) スタイルのユーザインタフェース
- 柔軟なプラグイン仕様
 - 情報技術者でなくても理解可能な直感的なデータ検索
 - 複数のデータソースを統合利用する場合でも個別のデータソースに係るシステム開発を必要としない
- 内部的に OGC の標準技術を利用する

これらの要件を実現するために rinzo.ma は図 2 の設計を持つ。rinzo.ma は Web ブラウザ上で Web ブラウザ上でいわゆる「リッチなインターネットアプリケーション (RIA)」を実現する ext.js と、マップ操作 API である Openlayers という 2 つのライブラリを利用して Web アプリケーションフレームワークを実装している。フレームワークは公開されたプラグイン仕様を持ち、また汎用的な機能、たとえば地図操作やジオコーディング等の機能は、あらかじめ標準プラグインとして提供されている。このフレームワーク上で、利用者は独自のデータや機能をプラグインとして追加することによって、目的に応じた Web アプリケーションを実現することができる。



図 2 rinzo.ma のコンポーネント
Fig. 2 Components of rinzo.ma.

これらの要件、設計に基づいて rinzo.ma は実装されている。次節では rinzo.ma の実装において利用している各種既存技術について述べる。

3.2.2 rinzo.ma が利用している既存技術

rinzo.ma は JavaScript で書かれている API と Web アプリケーションのひな型となる HTML ファイルで構成されている。利用者はひな型の HTML ファイルにプラグインを追加することで、独自の Web GIS アプリケーションを生成する。すなわち rinzo.ma は Web GIS アプリケーション生成のためのフレームワークという位置づけのソフトウェアである。

rinzo.ma は 2 つの既存 API を利用している。Web ブラウザ上でスクロール可能な地図機能を提供する Openlayers と、スタンドアロンアプリケーションと同等な GUI 部品を JavaScript アプリケーションで実現する ext.js である。

Openlayers は Google Maps のような Web ブラウザ上でスクロール可能なマップサービスを提供するための API で、Google Maps や Yahoo! Maps , Microsoft Virtual Earth のラップとして動作するほか、WMS , WFS 等の OGC で標準化されたマップサービスを利用することができる。rinzo.ma では衛星画像や GIS 画像の閲覧機能に Openlayers を利用している。データ提供者は OGC 標準、あるいはその他の Openlayers がサポートする形式でデータを提供することが条件である。

ext.js は、通常のスタンドアロンアプリケーションでは当然だが、HTML Form では実現しえないコントロール部品を JavaScript と Dynamic HTML の機能を使って実現している。rinzo.ma のフレームワークは ext.js を利用して Border レイアウトの RIA を実現している。また rinzo.ma プラグインは ext.js の有する豊富なコントロールを利用することができる。

JavaScript は、その文法としてはオブジェクト指向型プログラミングをサポートしていないが、prototype 型の言語であり、ある程度のコーディング制約によって簡易なオブジェクト指向を実現できる。Openlayers も ext.js も独自のオブジェクト指向を実現するためのコーディング制約を持っている。rinzo.ma では Openlayers の方式を採用してオブジェクトプログラミングを行っている。rinzo.ma が想定する利用者は、地球科学の研究者であり、Openlayers もその用途で使われているケースが多いため、利用者との親和性が、JavaScript エキスパートのための ext.js よりは高いと考えられる。

rinzo.ma はこれらの既存技術を使い、Web GIS アプリケーション実現のためのフレームワークと柔軟なプラグイン組み込みシステムを開発した。次項からは rinzo.ma に関して詳述する。

3.2.3 rinzo.ma の画面

図 3 は rinzo.ma 上で Google maps を表示したときの画面である。画面は Border レイアウトになっており、画面中央および東西南北に区切られている。CENTER パネルは地図を表示するために利用される。この図ではベースマップとして Google Maps が表示されている。この図では説明のために Google Maps の地図を表示しているが、ベースマップは選択的に利用でき、Google Maps が一部制限している商用利用等の用途では、独自のベースマップか、無償で公開されている Nasa の landsat WMS サーバ等を利用すればよい。

SOUTH パネルには rinzo.ma やそのプラグインが出力するログ情報が表示されている。ログは詳細なデバッグ情報を出力する DEBUG、アプリケーションの動作情報を出力する INFO、予期しない検索結果やエラーを表示する ERROR の 3 段階のレベルがあり、起動時にログレベル DEBUG を指定すると、DEBUG、INFO、ERROR すべてのログが表示され、INFO を指定すると、INFO、ERROR のみが表示される。プラグインの開発時と利用時でログレベルを切り替えて利用できる。

NORTH パネルには、トグルボタン（図中 1）とボタン（図中 2）が並ぶ。これらのボタンはプラグインによって定義される。各プラグインは 1 つのトグルボタンと複数のボタンを持つことができる。たとえば虫眼鏡のアイコンは地図のズーム機能を担当するプラグイン

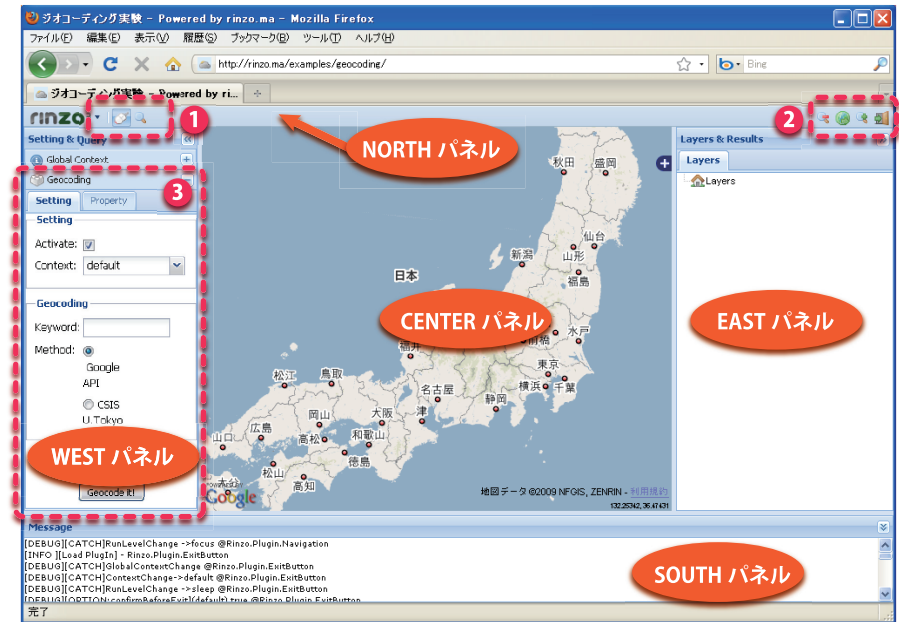


図 3 rinzo.ma のスクリーンショット
Fig. 3 Screenshot of rinzo.ma.

によって定義され、このトグルボタンを押下すると、CENTER パネル地図上に矩形領域を選択できるようになり、選択後その領域にズームする。また、NORTH パネル右にはズームイン・ズームアウト等のボタンが並んでいる。たとえばズームインのボタンを押すと、1 段階ズームレベルが上がる。

このように rinzo.ma では単に GIS データだけでなく、アプリケーション操作に関するあらゆる機能がプラグインとして実装されている。rinzo.ma プラグインは公開し共有・再利用することが可能なため、第三者による開発も期待できる。

EAST パネルは Global Property とプラグインが定義するパネルが配置されている。Global Property はプラグイン間でのデータ連係に利用するための共有データレジストリであり、詳細は次項で述べる。プラグイン関連パネルは、EAST パネルおよび WEST パネルに配置でき、WEST パネルはアコーディオン、EAST パネルはタブで、パネルの表示非表示を制御できる。これらのパネルはプラグインが自由にデザインできるが、WEST パネルは入力

表 1 日付と期間を表す ContextItem の例
Table 1 An example of ContextItem which represents a date range.

Key	Title	Value
DATE	日付	2009/9/20 0:0:0
RANGE	期間 (何日)	7

部, EAST パネルは検索結果等の出力部に利用することを推奨する。これにより, プラグインの開発者にかかわらず, 統一的操作感を提供できる。

図中では, Geocoding プラグインの入力画面が WEST パネルに表示されている。このプラグインはユーザーが入力した住所や地物の名前に応じた緯度経度を算出し, その地点が中心となるよう CENTER パネルの地図をスクロールさせる。利用者は Google Maps API と東京大学空間情報科学研究センターのジオコーディング Web サービスを選択的に利用できる。

3.2.4 Global Property

Global Property は rinzo.ma が読み込んだすべてのプラグインから読み書き可能な共有データレジストリである。Global Property に登録するデータを ContextItem と呼ぶ。ContextItem は Key, Title, Value の三つ組みからなる値のリストである。たとえば「2009年9月20日から1週間」という値を保持する ContextItem が保持するデータの例を表 1 に示す。このほか ContextItem にはデータを文字列に Serialize する機能, 文字列から内部形式へ Unserialize する機能を有する。現在は日付や地図上の矩形領域等の基本的な ContextItem を提供しているが, rinzo.ma が提供する ContextItem 実装のためのアブストラクトクラスをユーザーが継承し拡張することによって, 自由に ContextItem を増やすことができる。

rinzo.ma プラグインは Web アプリケーションに読み込まれている任意の ContextItem を購読することができる。購読した ContextItem はそのプラグインから書き換えることが可能になる。あるプラグインが購読している ContextItem を書き換えたとき, その ContextItem を購読している他のプラグインすべてに更新が通知される。日付をカレンダー形式で選択できるプラグインと, ある日付に撮影された衛星画像を地図上に表示するプラグインの連携を考える。両者がともに日付 ContextItem を購読していれば, カレンダープラグインでユーザーが選択した日付が, GlobalProperty の日付 ContextItem を更新し, その更新が衛星画像表示プラグインへ通知される。衛星画像表示プラグインは, その通知された日付に対応した衛星画像を表示することができる (図 4)。

このように, rinzo.ma では Global Property に対する Subscribe/Publish 処理でプラグ



図 4 ContextItem の Sub/Pub 処理
Fig. 4 Subscribe and publish of ContextItem.

インどうしの連携を図っている。

3.2.5 Web アプリケーションの構築

rinzo.ma フレームワークによる Web GIS アプリケーションの実装を, 実際のコードを例に説明する。図 5 は NASA が管理している MODIS センサ画像を表示するアプリケーションの JavaScript コード*1 である。

03 から 04 行目で rinzo.ma フレームワークのインスタンスを生成し Web アプリケーションの骨組みを生成する, その際, 引数として, 初期値を変更することができる。この例では, マップの表示領域を変更して, 初期状態で日本付近の地図が表示されるように設定している。

rinzo.ma のインスタンスを生成後, 読み込む ContextItem を追加する。08 から 09 行目で日付を保持する ContextItem を読み込んでいる。09 行目では, ContextItem の初期値を変更している。この例では日付の初期値として 2008 年 2 月 1 日を設定している。これによりこの ContextItem を購読するプラグインへは 2008 年 2 月 1 日という日付が通知される。たとえば衛星画像を表示するアプリケーションにおいては, その日時の衛星画像を検索・提示することが期待される。

*1 説明のため, 多少省略している。この説明と同等のアプリケーションは <http://rinzo.ma/examples/daily-modis/> で公開している。

```

01: Ext.onReady(function(){
02:   //rinzo.ma フレームワークインスタンスの生成
03:   var rinzo = new Rinzo({
04:     extent: {'west':120,'south':28,'east':136,'north':39}
05:   });
06:
07:   //ContextItem Date を追加 (DATE の初期値を 2008 年 2 月 1 日とする)
08:   rinzo.addItem(Rinzo.ContextItem.Date,{
09:     DATE:new Date(2008,2,1)
10:   });
11:
12:   // プラグインの追加 (rinzo.ma にバンドルされているプラグイン)
13:   rinzo.addPlugin(Rinzo.Plugin.Navigation);
14:   rinzo.addPlugin(Rinzo.Plugin.Zoom);
15:   rinzo.addPlugin(Rinzo.Plugin.GeoCoding,{forceZoomLevel:8});
16:   rinzo.addPlugin(Rinzo.Plugin.ExitButton);
17:   // ユーザ定義プラグインの追加 (MODIS 衛星画像閲覧プラグイン)
18:   rinzo.addPlugin(ac.yokoyama.Plugin.DailyModis);
19:
20:   //WEST パネルに GeoCoding プラグインの GUI を開く
21:   rinzo.macro.focusWest("Rinzo.Plugin.GeoCoding");
22:
23:   //EAST パネルに DailyModis プラグインの GUI を開く
24:   rinzo.macro.focusEast("Rinzo.Plugin.DailyModis");
25:
26:   //Navigation プラグインで定義された Navigation トグルボタンを押下
27:   rinzo.macro.clickGlobalToggle("Rinzo.Plugin.Navigation","Navigation",true);
28: });

```

図 5 Web GIS アプリケーションの JavaScript コード
Fig.5 JavaScript code of Web GIS application.

次に読み込むプラグインを追加する。プラグインには rinzo.ma があらかじめ用意しライブラリにバンドルしているプラグインと、第三者が独自に定義するプラグインがある。15 行目の GeoCoding プラグインでは forceZoomLevel 変数の初期値を 8 に変更している。これにより、ジオコード結果をマップに表示する際、そのズームレベルは強制的に 8 となる。

20 行目以降は、画面の初期状態の定義を行っている。WEST パネルと EAST パネルにはプラグインが定義したパネルを選択的に表示することができるが、どのプラグインに関連したパネルを表示するのかを定義している。また 27 行目で、初期状態では Navigation プラグインが定義するトグルボタンが押下された状態を初期状態と定義している。

この図では JavaScript コードのみを記しているが、実際にはこのコードは HTML ファイル中の <head> タグ内に配置され、他に JavaScript ファイル、CSS ファイルの読み込み等を必要とする。しかしながら、rinzo.ma のコードは非常に構造的であり、機械的に生成することが可能であるため、実際の利用者はこの HTML ファイルや JavaScript コードすら記述する必要がない。今後の課題であるが、我々は利用者がプラグイン一覧から任意のプラグインを選び、コードを自動生成する手法を検討している。

3.2.6 アプリケーションの状態保存

Web アプリケーションは構造的に動作状態の保存が困難であるといわれている。内部状態を保存するには Cookie の限られた領域に文字列として保存するか、1 度 Web サーバへ内部状態を渡しファイルを生成してサーバからダウンロードするという手法がよく使われている。前者は保存したデータが Web ブラウザにより予期せず削除されてしまう場合がある等永続的な保存には適さない。また後者は、1 度全セーブデータを 1 つのサーバに渡す必要がある。rinzo.ma がターゲットとする GIS データは、非常に高価なものあるいは機密性の高いものも多く、そのようなデータがセーブデータを生成するために 1 つのサーバへ集約することは現実的な設計とはいえない。

rinzo.ma は、簡易的な保存手法として Cookie への保存機能は有するが、既存する手法とは異なる URL 形式と JSON 形式での保存方法を採用している。なお rinzo.ma により保存されるデータは、保存した時点における各 ContextItem のデータ、プラグインのシリアルライズ情報、画面の状態等である。URL 形式で保存した場合はこれら保存情報を QueryString として有する 1 つの URL がユーザに提示される。セーブデータは URL 形式なので Web ブラウザのブックマークに保存することができる。利用者はいつでもブックマークから作業を再開することができる。

ファイルとして保存したい利用者のためにセーブデータを JSON 形式のテキストとして表示する機能も有している。利用者はその JSON テキストをコピーし任意のファイルに保存することができる。Firefox においては JavaScript において直接 JSON ファイル生成し、ダウンロードする機能があるため、コピー&ペーストなしにセーブデータをファイルとしてダウンロードすることが可能である。

4. rinzo.ma プラグイン

4.1 プラグインの概要

rinzo.ma プラグインは JavaScript で実装され、必要に応じて Ajax 等の非同期通信技術

を用いて外部サーバーと通信する．画面上へ表示する GUI 部品には ext.js API を利用することができる．また CENTER パネルのマップを Openlayers API を使って操作することができる．プラグインは少なくとも 1 つの JavaScript ファイルからなる．GUI 部品としてアイコン等を用いる場合は、その画像ファイルと、アイコン表示のための CSS ファイルを含むことができる．またサーバ側の処理を必要とするときは PHP ファイルを含むことができる．

4.2 Context

プラグインには動作の複数のルーチンを定義することができる．このルーチンを Context と呼ぶ．また Context ごとに購読する ContextItem を定義することができる．たとえば、衛星画像検索のプラグインを考えると、時間で検索する Context、マップ上の矩形範囲で検索する Context 等、問合せごとに Context を定義する．時間で検索する Context では日付の ContextItem を購読し、矩形範囲で検索する Context では、矩形範囲を表す ContextItem を購読すればよい．地図上に矩形範囲を描くプラグインやカレンダーによって日付を指定できるプラグインと一緒に読み込むことにより、GUI 操作で衛星画像の検索を行うことができる．

4.3 Run Level

プラグインは Sleep, Activate, Focus という 3 つの Run Level を持つ．Run Level が Sleep のときはそのプラグインは何も処理を行わず、また購読している ContextItem に変更があった場合でも通知されない．Activate と Focus はともに、プラグインが動作している状態のことを表す．ただし Run Level Focus が与えられるのは、読み込まれているプラグインのうち 1 つだけである．排他的な処理を Focus に割り当てることを想定している．Run Level の変更はユーザが明示的に指定できるほか、プラグインが自らの Run Level を変更することもできる．

このように、rinzo.ma プラグインは定義した Context と Run Level のマトリックスをつくり、その各セルにおいて、プラグインの動作を定義するというモデルを持つ．

4.4 ボタンとパネル

プラグインには 2 つのパネルと任意の数のトグルボタン、ボタンを定義できる．定義された GUI 部品は rinzo.ma によって、画面内に適切に配置される．

図 6 は、rinzo.am によって配置されたパネルの例である．この例では WEST パネルにジオコーディングのプラグイン、EAST パネルには ASTER 衛星画像アニメーションプラグインのパネルが最前面に表示されている．どのプラグインのパネルを表示するかは、3.2.5 項で述べたように HTML で定義できるほか、実行時にユーザが表示するパネルを切り替える

NORTH パネル



ジオコーディングプラグイン

ASTER 衛星画像アニメーションプラグイン

図 6 rinzo.ma によって配置されたプラグインのパネルとボタン
Fig. 6 Panels and buttons which is located by rinzo.ma.

ことができる．

NORTH パネルには左側にトグルボタン、右側にボタンが配置される．トグルボタンは CENTER パネルのマップに対する排他的な処理の切替えに使う．たとえばマウスを地図上でドラッグしたときに、スクロールをするのかズームをするのかを利用者は選択できる．排他的ではない処理、非同期で走るバックグラウンド処理は右側のボタンに定義する．

マップ上での操作を NORTH パネルに、プラグインへの入力を WEST パネルに、プラグインからの出力を EAST パネルに配置することによって、複数のプラグインを利用しても、共通の操作感を利用者に提供することができる．

図の例では、WEST パネルと EAST パネルにそれぞれ別のプラグインが提供するパネル

が表示されている。この Web GIS アプリケーションは、地物名を入れるとその地物が存在する場所の過去から現在までの衛星画像を検索し、時系列順にアニメーション表示をする。また雲のかかった画像等を利用者が排除する仕組みも持っている。利用者は、まず WEST パネルに表示されているジオコーディングプラグインのパネルに、観察をしたい場所を入力する。その情報は Google Maps のサーバに送られ、その場所の緯度経度が求められる。その情報は ContextItem を通じ、それを購読している衛星画像アニメーションプラグインへ通知される。衛星画像アニメーションプラグインは Ajax を使い、産業技術総合研究所の GEO Grid プロジェクトが有する ASTER 衛星画像のメタデータサーバへ問い合わせ、その場所の時系列衛星画像へアクセスするための URL を得る。衛星画像アニメーションプラグインは Openlayers API を使い、その時系列衛星画像を CENTER パネルに表示し、アニメーションを行う。このように、複数のプラグインを連携して地物の名称や住所から、その場所の衛星画像をアニメーション表示するというワークフローを、検索された衛星画像の取捨選択するユーザインタラクションを阻害しない形で実現している。

図の例では、ジオコーディングプラグインへ「静岡空港」と入力しているので、2009年6月4日に開港した静岡空港が建設される様子をアニメーションで表示することができる*1。利用者は静岡空港と入力し、サブミットボタンをクリックするだけで、静岡空港の位地や衛星画像とそのメタデータに関する知識がなくても、必要な衛星画像を検索し表示することができる。

4.5 Ajax によるサーバとの通信

プラグインは JavaScript で書かれ Web ブラウザ上で実行され、必要に応じてサーバと非同期通信 (Ajax) を行う。Ajax とは Web ブラウザが有する XMLHttpRequest クラスを使った通信のことである。Ajax にはセキュリティ上の制約があり、その JavaScript プログラムを呼び出した HTML ファイルを保持する Web サーバとしか通信できない。rinzo.ma のモデルでは、データ提供者がプラグインも提供する。一方で rinzo.ma フレームワークに基づく Web GIS アプリケーションの HTML ファイルはデータ利用者の有するサーバ上に配置される。Ajax 通信の制約によりプラグインはデータ提供者の有するサーバとは通信できない。そこで、我々はこの制約を回避する 2 つの方法を提供している。

1 つは、データ利用者のサーバ上を経由して、プラグイン/データ提供者のサーバと通信

する方法である。rinzo.ma の実装にはこの Ajax 通信を中継するためのプロキシが含まれている。そのプロキシに対し、プラグインが中継ルールの定義を与えることにより、プラグインは、データ利用者のサーバを中継し、プラグイン/データ提供者のサーバ、あるいは第三者のサーバと通信することができる。

このプロキシを使う方法は Web アプリケーションにおいては一般的な手法である。しかしながらインターネット上に広く分散した衛星データや地質データ等 GIS データをオンデマンドで統合する処理を考えると、プロキシを使う方法は必ずしも現実的ではない。なぜならば、GIS データは高価かつ機密性の高いデータも含まれ、そのユーザ操作がデータ提供者以外のサーバを経由することは、セキュリティの面で問題となる可能性がある。

Web ブラウザから直接第三者のサーバへアクセスする手法としては、我々が開発する FREDDY⁹⁾ がある。FREDDY は Web ブラウザと任意のサーバ間で半構造データのストリーム通信を実現する手法であり、これを利用すれば、プラグインは直接プラグイン提供者のサーバと通信することができる。

このように、rinzo.ma では Web ブラウザ上のプラグインと、サーバ上の処理プログラム間の通信手法として、プロキシを介した Ajax 通信、そして FREDDY という 2 つの手法を提供している。

4.6 マップ操作

次にプラグインによるマップ操作について説明する。CENTER パネルに表示されるマップは Openlayers によって描画されおり、プラグインはそのマップに対し独自のレイヤを追加することができる。定義可能なレイヤは Openlayers に準ずる。すなわち OGC の標準化されたサービス WMS や WFS が利用可能である。また同時にプラグインでは各 Context, Run Level において、そのレイヤを描画するかどうかを指定する。rinzo.ma はその指定に応じて、レイヤの描画・非描画を決定する。たとえばすべての Context において、Run Level Sleep のときはそのプラグインが定義する全レイヤを描画しない、といった設定が可能である。

4.7 プラグインの従属関係

プラグイン間でのデータ交換は Global Property を介して行うことはすでに述べた。しかしながら、この方法だけでは簡単な文字列や数値の交換には適しているが、別のプラグインのメソッドを呼び出したり、GUI を利用したりといった、複数プラグインの協調動作を実現することが困難である。rinzo.ma では、あるプラグインを読み込む際に、同時に読み込まなければならないプラグインを指定することができる。これをプラグインの従属関係

*1 このアニメーションを GIF アニメーション化したものを http://rinzo.ma/miya/shizuoka_airport.gif で公開している。

(dependancy) と呼ぶ。

たとえば衛星画像検索プラグインがジオコーディングプラグインの利用を前提としている場合、衛星画像検索プラグイン内で、ジオコーディングのプラグインのクラス名を従属関係のあるプラグインとして定義する。これにより衛星画像検索プラグインからジオコーディングプラグインのメソッド呼び出しが容易^{*1}になる。

この仕組みにより現在の実装においては、従属関係にあるプラグインが読み込まれていないとき、その旨のエラーを出力している。また、現在取り組んでいる実装においては、従属関係にあるプラグインが読み込まれていない場合、そのプラグインの JavaScript ファイルを自動的にダウンロードし、アプリケーションに組み込む手法を検討している。

5. rinzo.ma を使った GIS アプリケーション

本章では、これまでに述べた Web GIS アプリケーションフレームワークを利用している実際のアプリケーションについて述べる。

5.1 MODIS による黄砂観測

MODIS はアメリカ航空宇宙局 (NASA) によって開発されたセンサであり地球観測衛星 Terra, Aqua に搭載されている。MODIS の解像度は 500 メートルであるものの、毎日地球全域を観測できる。この MODIS のデータの一部は NASA のジェット推進研究所 (JPL) が OGC WMS 形式で公開している。我々はこの公開されているデータを rinzo.ma 上に表示するプラグインを開発した。それを使い rinzo.ma のデモンストレーションのために、2008 年の黄砂飛来状況を観測する Web GIS アプリケーションを実装した。

図 7 はその Web GIS アプリケーションによって観察された、黄砂の飛来状況である。rinzo.ma を使えば、このような時系列画像を順に閲覧する等の処理を簡単に実現することができる。図は 2008 年 3 月 1 日から 4 日までの中国から西日本付近を撮影した MODIS 画像である。3 月 1 日に山東半島上空にある黄砂が、2 日間かけて日本上空へ飛来し、3 月 4 日には黄砂が画像内から消えていることが観察できる。

MODIS をはじめとして衛星画像はデータ量が膨大であり、ただ公開されているからといって、簡単に利用することはできない。公開するデータを、データ統合・処理・利用のためのフレームワーク上で GUI も含めて共有することにより、GIS データの効果的な活用が

*1 JavaScript ではメソッドの呼び出しに制限を付けることができないため、従属関係のないプラグイン間のメソッド呼び出しを禁止することはできない。

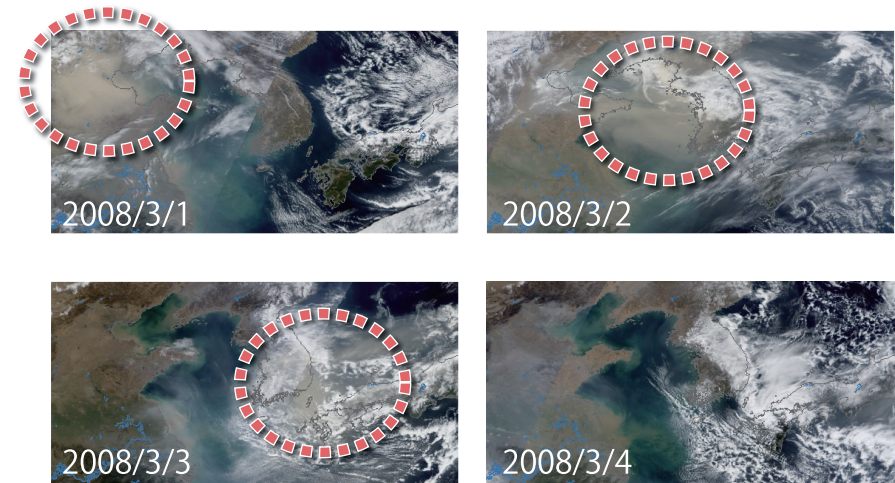


図 7 Terra/Modis センサによる黄砂観測
Fig. 7 Asian Dust observed by TERRA MODIS.

期待できる。

5.2 ASTER 衛星画像のモザイク生成と時系列アニメーション

ASTER は MODIS も搭載する地球観測衛星 Terra に搭載されているセンサであり、全球のデータを 15 メートルの解像度で収集している。産業技術総合研究所の GEO Grid では 2000 年 3 月 4 日から 1 日平均で約 470 シーン、全 160 万シーンの画像およびメタデータを、グリッド技術を使いハードディスク上で保持している。観測された生データからメタデータを生成し可視化するのに約 4 日間のタイムラグがあるが、毎日、新しく観測された画像が増え続けている。問題は、これだけ莫大な数、サイズのデータをどのように検索するかである。

しかも衛星画像はパラメータによる検索だけでは、有用な画像を効率的に見つけることはできない。たとえば、ASTER のデータを元に画像処理技術を使い、各シーンの中に雲と思われる画素を判定し、その割合をメタデータとして付けたとする。その割合が 10% 以下のシーンを探せば、晴れている日の画像を検索できるかもしれない。しかしながら雲の割合が 10% 以下だからといって、その画像中の観察したい場所に雲がかかっていないとは保証できない。かといって画像中の雲の場所までもメタデータとして持つことは、データ管理・

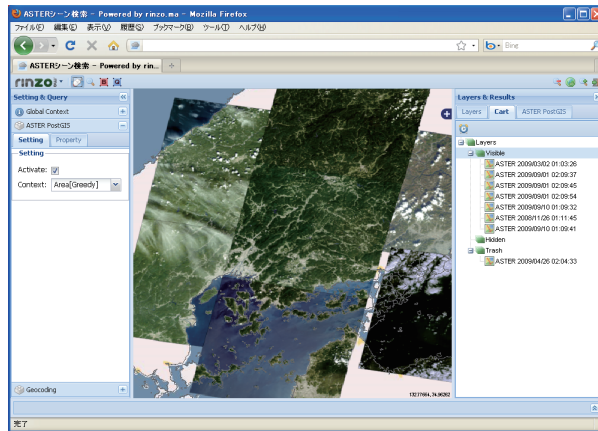


図 8 ASTER 画像のモザイク生成
Fig. 8 Creating ASTER mosaic image.

検索のスケラビリティの点からも現実的ではない．この例のように地球観測においては利用者が実際にデータ見て判断するというフェーズが必ず発生する．

図 8 は，rinzo.ma を使って実装した ASTER の衛星画像のモザイク生成アプリケーションである．このアプリケーションは産業技術総合研究所においてデジタル標高マップ (DEM) 生成のためのユーザインタフェースとして利用されている．この GIS アプリケーションはショッピングカートのメタファに基づいたツールである．利用者は日付範囲や地理的範囲等の条件を rinzo.ma の GUI で指定することにより，システムは ASTER のメタデータデータベースから条件に合ったシーンを利用者に提示する．そして利用者はそのシーンが DEM 生成に適しているかを判断し，適切だと判断した画像をショッピングカートに入れる．ショッピングカートに入れた複数のシーンをどの順番で重ねるかを GUI を通して指定し，最後に注文ボタンを押すことで，GEO Grid サーバ上の DEM 生成プログラムへ，選択されたシーンとその重ね順が通知される．

ASTER のメタデータデータベースは PostgreSQL/PostGIS で管理している．この DEM 作成時に rinzo.ma から利用されるメタデータは観測日時，シーンの撮影範囲 (PostGIS の Polygon 型で管理)，雲量 (画像処理によって算出) である．このアプリケーションでは ASTER 画像検索プラグインと画像ショッピングカートプラグイン，矩形範囲指定プラグイン，ジオコーディングプラグインを使用している．ASTER 画像検索プラグインはユーザの

表 2 ASTER 画像検索プラグインの Context
Table 2 Contexts of ASTER image serch plug-in.

Context	説明	購読 ContextItem
Date	日付で検索	Data, CloudCoverage
Date[Range]	日付の範囲で検索	DateRange, CloudCoverage
Area[Strict]	矩形範囲に含まれる/含む画像検索	QBOX, CloudCoverage
Area[Greedy]	上記に加え矩形範囲に触れる画像も検索	QBOX, CloudCoverage
Complex	全条件による検索	DateRange, QBOX, CloudCoverage

検索モードに応じた 5 つの Context を持っている (表 2)．利用者はこの 5 つの検索方法を切り替えながら目的の衛星画像を検索し取捨選択しながらモザイク画像を生成することができる．

ASTER のような中・高解像度のデータを取得するセンサの場合，1 度に撮影できる範囲は限られている．そこで，このように別の時点で撮影されたデータを含めモザイク状に画像を並べることにより，広範囲の高解像度画像を得ることができる．

同じく ASTER を使い，同地点を撮影した画像を複数検索し，それを時系列順でアニメーション表示するアプリケーションも我々は開発している．前述したモザイク生成アプリケーションは地理的に隣接した画像を並べることによって，広範囲にわたる高解像度衛星画像およびその範囲の DEM を生成できたが，このアニメーション生成アプリケーションは撮影日時が時間的に隣接している画像を順次閲覧することによって，同地点の変化を見ることができる．

図 9 はこのアニメーション生成アプリケーションにより観察したつくばエクスプレス建設の様子である．この画像はつくばエクスプレスの守谷駅，みらい平駅間の水田地帯で，rinzo.ma の名前の由来になった間宮林蔵の生家の付近である．2000 年 9 月 21 日の画像では水田だった場所に，2002 年 3 月 19 日以降の画像には，筑波エクスプレスの高架橋ができあがっていく様子が現れている．さらに 2009 年 2 月 2 日の画像ではみらい平駅付近に建物が多く建っていることが分かる．

このアプリケーションから利用されている ASTER 検索プラグインは，ある地点の緯度経度の入力があると，その地点を含んでいる衛星画像をメタデータデータベースから検索し，画像中の雲量が 10% 以下の画像のみを返す．しかしながら，前述したようにこのような単純検索だけでは，2007 年 5 月 11 日の画像のように，雲量は 10% 以下であるものの雲が観察箇所にかかっているデータは排除できない．このような画像を利用者が実際に見て排除する仕組みも備えている．

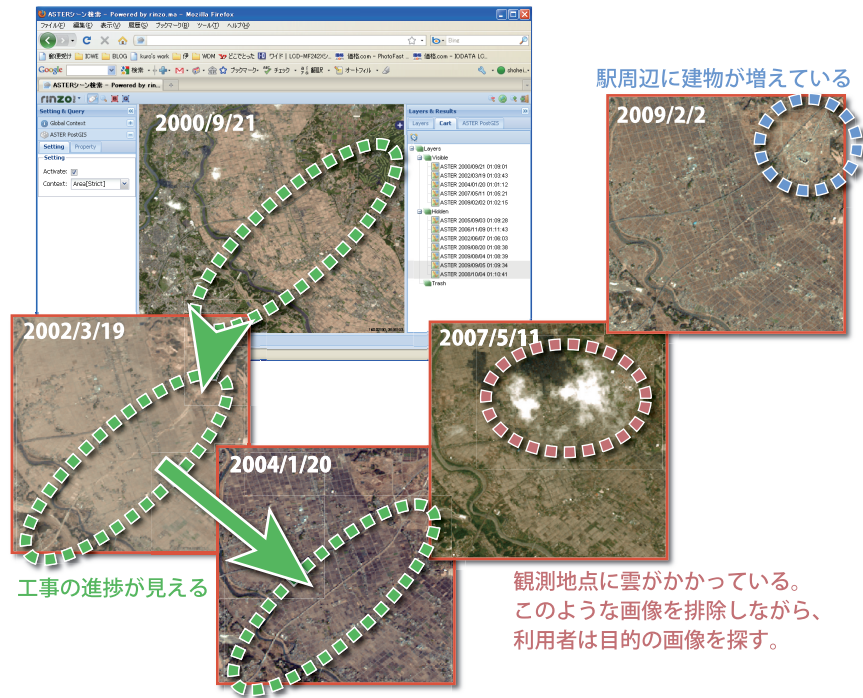


図 9 ASTER 画像によるつくばエクスプレス沿線観測
Fig. 9 ASTER images along the TSUKUBA EXPRESS.

本節で紹介したアプリケーションは場所や時間を入力することにより、ASTER データの巨大なデータベースから適切な衛星画像を検索する仕組みである。実際には利用者は場所や時間を指定して画像を検索したいのではなく、「つくばエクスプレス建設の様子を検索したい」という事象についての検索するという文脈において利用されることが想定される。そこで、我々は時間や場所を直接入力するのではなく、事象を入力することによって衛星画像を検索するシステムを研究開発した。このシステムを次節で詳述する。

5.3 地震前後の衛星画像による被害状況の確認

本節で紹介するアプリケーションは、大規模地震の一覧から 1 つ地震を選べると、その地震の震源付近でかつ地震発生前後の画像を検索しアニメーション表示する。大規模地震の一覧はあらかじめ用意する必要があるが、Wikipedia から自動抽出している。この手法、アプリ

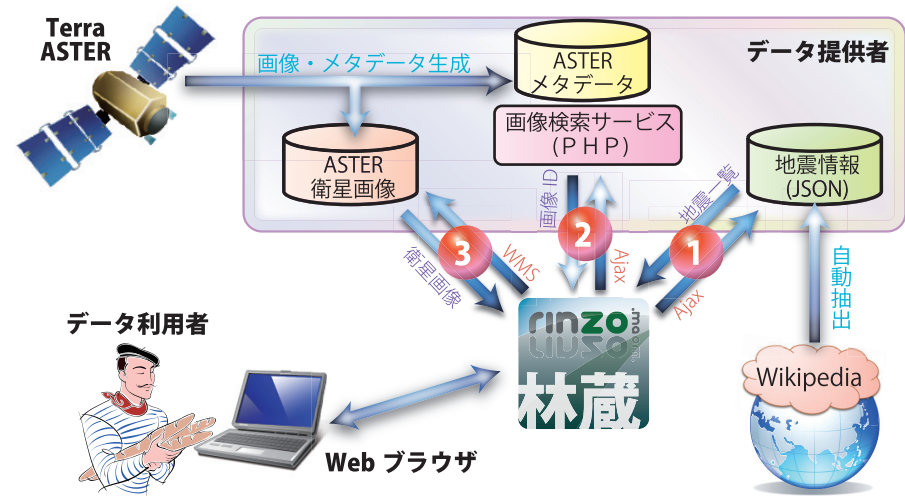


図 10 地震被害観察アプリケーションのシステム外観
Fig. 10 Outline of the earthquake damage browser.

ケーションに関しては文献 11) に詳しい。本論文では、そのアプリケーションにより観察された岩手・宮城内陸地震（2008 年 6 月 14 日発生）による山体崩落現場の画像を紹介し、アプリケーションの概要を説明する。

システムの概要を図 10 に示す。衛星画像は GEO Grid プロジェクトによる ASTER メタデータ・画像データを利用する。地震に関するデータは我々が開発した Wikipedia からの情報抽出手法¹⁰⁾を用いる。この手法では Wikipedia の構造化された文書を解析することによって、記事に付与されている重要な日時と場所を抽出することができる。たとえば地震のページにはその地震に関するデータ、発生日時や震源の位置情報が infobox という機能を用いて構造的に記述されている。また各記事は詳細にカテゴリ化されており、地震に関する記事は 1 つのまとまったカテゴリからたどることができる。このような Wikipedia の特徴を利用して、地震に関する情報を抽出している。

図 10 には、Wikipedia から抽出したデータを使って衛星画像を得るためのワークフローが記されている。①Wikipedia から抽出された地震情報は JSON 形式のファイルとして Web サーバ上で公開されている。そのデータを読み込みユーザへ地震の一覧を提示する。地震一覧はたとえばマグニチュードで並べ替えたりといった、最低限の地震検索機能を持つ。②

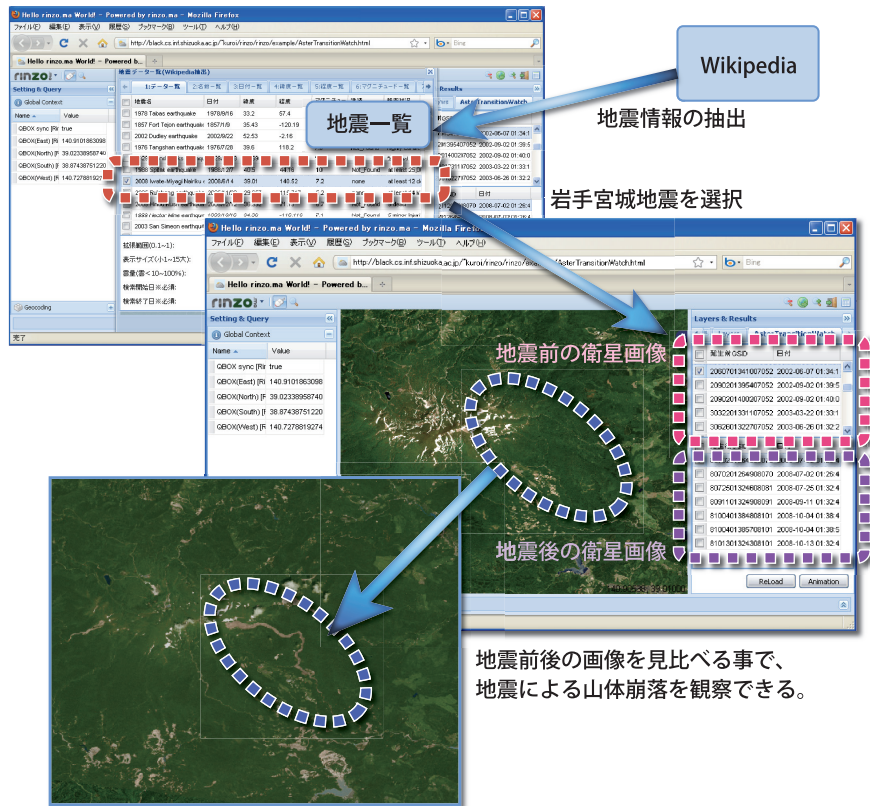


図 11 岩手・宮城内陸地震による被害
Fig. 11 Damage of the Iwate-Miyagi Nairiku earthquake.

ユーザが 1 つの地震を選ぶと、その発生時刻と震源の場所が ASTER のメタデータサーバへ通知され、震源付近における発生前後に撮影した衛星画像の識別子が得られる。③得られた識別子に基づいて ASTER の WMS サーバから衛星画像を表示する。最後に、時系列順でアニメーション表示をする。

図 11 は、このアプリケーションで生成した岩手・宮城内陸地震による山体崩落現場の地震前後の画像を示している。利用者はアプリケーションの画面に現れる地震一覧から岩手・宮城内陸地震を選ぶだけで、その地震前後の画像を閲覧することができる。このアプリケー

ションの特徴的な点は、データ提供者が地震が起きたからといって、その様子が観測されている画像を自ら用意して提示する必要がないこと、そして利用者はその地震に関する詳細情報を持っていなかったとしても、簡単な画面操作だけで、衛星画像が検索できること、またシステムの開発者には地震に関する知識は求められず、それらの分野固有の知識を、この例では Wikipedia という集合知にアウトソースしていることである。GEO Grid 等の e-science に関連する研究では、Science 諸分野の研究者と情報技術の研究者が協力して研究を進めているが、情報技術の研究者には科学諸分野固有の知識を前提とするのは困難であり、また Science 諸分野の研究者にとっても情報技術に関する知識を前提とするのは難しい。本節で紹介したアプリケーション例はこの両者を結び付けるためのフレームワークとして rinzo.ma が利用できることを示している。

6. まとめ

本論文では、Web GIS アプリケーションフレームワーク rinzo.ma について、その設計と応用について述べた。rinzo.ma は柔軟なプラグイン仕様を持ち、容易に共有・再利用を行うことができる。また複数のプラグインを組み込むことにより、GIS データに対するワークフローを実行できる。rinzo.ma はすでに GEO Grid プロジェクトにおいてデジタル標高マップのモザイク生成のインターフェースとして利用されており、また、ASTER だけでなく、MODIS 等の他の衛星画像を扱うシステムも紹介した。さらに、Wikipedia から情報を抽出するツールを用いて災害情報閲覧を容易に実現するシステムの実装について述べた。

今後の課題としては、第三者にプラグイン仕様の詳細を公開するとともに、プラグインを登録するリポジトリの構築を考えている。プラグインの再利用を促進し、実行時における動的な読み込みやプラグインどうしの連携の仕組み等を実現する。

謝辞 論文中の ASTER 画像データは産業技術総合研究所 GEO Grid プロジェクトからデータの提供を受けている。ここに感謝の意を表す。また本研究の一部は科学研究費補助金基盤研究 (A) (課題番号 20240010)、若手研究 (B) (課題番号 20700104) の助成による。

参考文献

- 1) International Virtual Observatory Alliance. <http://www.ivoa.net/>
- 2) IVOA Virtual Observatory Query Language Working Group, howpublished=.
- 3) OGC Web Services-Phase 4. <http://www.opengeospatial.org/pub/www/ows4/index.html>
- 4) Open Geospatial Consortium, Inc. (OGC). <http://www.opengeospatial.org/>

- 5) Justice, C., Vermote, E., Townshend, J.R.G., Defries, R., Roy, D.P., Hall, D.K., Salomonson, V.V., Privette, J., Riggs, G., Strahler, A., Lucht, W., Myneni, R., Knjazihhin, Y., Running, S., Nemani, R., Wan, Z., Huete, A., van Leeuwen, W. and Wolfe, R.: The Moderate Resolution Imaging Spectroradiometer (MODIS): Land remote sensing for global change research, *IEEE Trans. Geoscience and Remote Sensing*, Vol.36, No.4, pp.1228–1249 (1998).
- 6) Lin, K. and Baru, C.: Geosciences data integration, *GEON Cyberinfrastructure Workshop*, Beijing, China (2006).
- 7) Sekiguchi, S., Tanaka, Y., Kojima, I., Yamamoto, N., Yokoyama, S., Tanimura, Y., Nakamura, R., Iwao, K. and Tsuchida, S.: Design Principles and IT Overview of the GEO Grid, *IEEE Systems Journal*, Vol.2, No.3, pp.374–389 (2008).
- 8) Yamaguchi, Y., Kahle, A.B., Tsu, H., Kawakami, T. and Pniel, M.: Overview of Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER), *IEEE Trans. Geoscience and Remote Sensing*, Vol.36, No.4, pp.1062–1071 (0000).
- 9) 横山昌平: Web ブラウザから利用できる SAX パーサ “Freddy” の実装と評価, 電子情報通信学会論文誌 D, Vol.J91-D, No.3, pp.585–594 (2008).
- 10) 岡本章裕, 黒井星良, 横山昌平, 福田直樹, 石川 博: Wikipedia を対象とした地理情報と時間情報の抽出手法の提案, DEIM 2009: 第 1 回データ工学と情報マネジメントに関するフォーラム, 静岡県, 日本, pp.1–8 (2009).
- 11) 黒井星良, 岡本章裕, 横山昌平, 福田直樹, 石川 博: Web 技術を利用した効果的な災害前後の衛星画像検索・閲覧基盤の開発, DEIM 2009: 第 1 回データ工学と情報マネジメントに関するフォーラム, 静岡県, 日本, pp.1–8 (2009).

(平成 21 年 9 月 18 日受付)

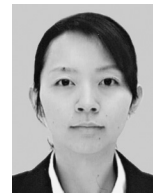
(平成 21 年 11 月 17 日採録)

(担当編集委員 井上 潮)



横山 昌平 (正会員)

静岡大学情報学部助教。産業技術総合研究所を経て 2008 年より現職。2006 年東京都立大学大学院工学研究科修了, 博士 (工学)。データベース技術の研究開発に従事。電子情報通信学会, 日本データベース学会正会員。情報処理学会論文誌データベース編集委員 (幹事補佐), 電子情報通信学会データ工学研究会幹事補佐。



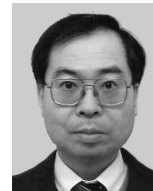
黒井 星良

2009 年静岡大学情報学部情報科学科卒業。同年スズキ教育ソフト株式会社入社。現在に至る。地震前後の ASTER 衛星画像検索システムの研究開発に従事。



岡本 章裕

2009 年静岡大学情報学部情報科学科卒業。同年同大学大学院情報学研究科修士課程入学。現在に至る。Wikipedia からの時空間情報抽出に興味を持つ。日本データベース学会学生会員。



石川 博 (正会員)

静岡大学情報学部情報科学科教授。東京大学理学部情報科学科卒業。東京都立大学を経て 2006 年より現職。東京大学博士 (理学)。著書に『次世代データベースとデータマイニング—DB & DM の基礎と Web・XML・P2P への適用』(CQ 出版社), 『JavaScript によるアルゴリズムデザイン—オブジェクト指向から DB・Web・マイニングまで』(培風館), 『データベース』(森北出版)等。国際論文誌 ACM TODS, IEEE TKDE, 国際学会 VLDB, IEEE ICDE 等学術論文多数, 1994 情報処理学会坂井記念特別賞, 1997 科学技術庁長官賞 (研究功績者) 受賞。情報処理学会データベースシステム研究会主査, 情報処理学会 (データベース) 共同編集委員長, International Journal Very Large Data Bases Editorial Board, 日本データベース学会理事歴任。ACM, IEEE 各会員。