

ユーザーの所持するデータベースを利用可能な 考古学データ検索システム

辻 一明[†] 宝珍 輝尚[‡] 野宮 浩揮[‡]

[†] [‡] 京都工芸繊維大学工芸科学部 〒606-8585 京都市左京区松ヶ崎御所海道町

E-mail: [†] yuebito319@yahoo.co.jp, [‡] {hochin, nomiya}@kit.ac.jp

あらまし 本稿では、遺跡から多数出土する遺物情報を管理する考古学データ検索システムの設計と実装について報告する。ここでは、ユーザーにとって使いやすく便利なシステムを目的とし、クライアント側でユーザーの所持するデータベースに接続し、検索をすることができ、さらにサーバーに検索結果を送信することにより、GoogleMap上に遺物の出土分布表示を可能にするような、デスクトップ上で動作するデータベースアプリケーションの実現を行った。クライアント側でデータベース接続と検索を行うため、ユーザーの所持するデータベースをサーバーへコピーする必要が無いことが、特徴となっている。また、検索前にビューを作成するため、ユーザーのデータベースをシステム側に合わせる必要もなくなっている。

キーワード 考古学データベース, 分布表示, ビュー

Archeological Data Retrieval System Using User's Databases

Kazuaki TSUJI[†] Teruhisa HOCHIN[‡] and Hiroki NOMIYA[‡]

[†] [‡] Undergraduate School of Information Science, Kyoto Institute of Technology, Goshokaido-cho, Matsugasaki, Sakyo-ku, Kyoto-shi, Kyoto 606-8585 Japan

E-mail: [†] yuebito319@yahoo.co.jp, [‡] {hochin, nomiya}@kit.ac.jp

Abstract This paper studies on the design and implementation of archaeological data retrieval system that manages information of many relics excavated from ruins. For the purpose of constructing useful system for users, we developed a desktop database application that can show a distribution of excavated relics on Google Maps by connecting to and searching a database on the client, and by sending search results to the server. In this system, we connect to and search a database on the client. It is unnecessary for us to copy their own databases to the server. We create “view” before searching a database. We do not have to adjust our databases to the system database.

Keyword Archeology Database, Distribution of Search Results, View

1. はじめに

近年のコンピュータ技術の発展はめざましく、考古学に関するデータを電子媒体として、コンピュータ上で取り扱っている考古学者も少なくない。遺跡データや遺物データといった考古学データは、その特性から、よく地理情報と関連付けて保存されている。そういったデータをコンピュータ上で管理するためには、ユーザー自身が地図データを用意し、利用する必要がある。そのため、Web上で公開されている地図サービスを利用することで、ユーザー自身が地図データを用意する必要がないシステムが提案されている。しかし、こういった従来の考古学データ検索システムを利用するためには、ユーザーの所持するデータベースを、自分の手が完全に行き届かないサーバーなどに保存する必要があることが多い。

考古学者にとって、自分の所持する研究データは大変貴重なものであり、安易に手の届かないところに置

きたくないのが現状である。そのため、これまで提案されてきているシステムは、魅力的なものではあるが、自分の所持するデータベースをサーバー側へコピーすることは、考古学者にとって躊躇われるものである。

そこで、本研究では、ユーザーの所持するデータベースを、サーバー側へコピーすることなく利用可能な考古学データ検索システムの実現を目的として、検討を行う。このために、クライアント側で検索を行い、その集約検索結果のみをサーバー側に送信することで、地図上に遺物の出土分布の表示が可能なシステムを実現する。本研究と従来の研究との違いは、クライアント側で、ユーザーの所持する考古学データベースへの接続、検索を行っているため、サーバー側にデータベースをコピーする必要がないことであり、この点が、本研究の一番の特徴となっている。したがって、クライアント側で、ユーザーの所持する考古学データベースに対しての処理を行うような、デスクトップアプリ

ケーションが必要となる。本研究では、それらをアドビ・システムズ社が開発した Adobe AIR というランタイムライブラリを用いて実現している。

本論文では、関連研究や対象とする考古学データ、本システムにおけるクライアント側プログラムとサーバー側プログラムの設計と実装、その動作例を報告する。

以下、まず 2. で関連研究について述べる。次に、3. で対象とするデータについて説明する。そして、4. で使用するシステムを概説する。5. で具体的な設計について述べ、6. で実装について述べる。7. で動作例を示し、最後に 8. でまとめる。

2. 関連研究

ここでは、これまでに研究されてきた様々な考古学データベースシステムに関する研究などについて概説する。

2.1. goo 地図を利用した遺跡データベースシステム

山内らは、遺跡から多数出土する遺物を地図情報と連携して管理する遺跡データベースシステムの設計と実装について報告している[1]。管理者にも一般の利用者にも使い易い考古学データベースシステムの実現を目的として、一乗谷朝倉氏遺跡の遺物データをもとに、遺跡から多数出土する遺物を、地図情報と連携して管理する遺跡データベースシステムの設計と実装を行っている。

従来では、地図情報は管理者自らが用意していたが、Web 上で提供されている地図サービス「goo 地図」を利用することで、地図情報の管理の簡便化を図っている。また、遺跡と遺物、遺構の上下関係をもとに、これらのデータ実体と、分布表示における「全体」「集約単位」「検索対象」を表すそれぞれのデータ実体を対応づけることで、様々な分布表示を可能にするために、各クラスの上下関係を念頭においてデータベースの設計が行われている。

2.2. 考古学データベースシステムにおけるユーザーデータベースの利用

小林らは、考古学データベースシステムにおいて、従来ではユーザーがシステムに自身のデータベースを導入する際に、ユーザーが所有しているデータをシステムに沿った形で登録する必要があり、その手順が非常に面倒であると考えた。また、遺跡や遺物の情報は、その特性上、所有者が一般ではなく、一部に限定的に公開したいものであると考えて、ユーザーが遺物データを公開したい相手のみに対して検索可能な状態にできるような考古学データベースシステムの設計を行っている[2]。システム内に、ユーザーのデータベース接続に必要な情報を保存しておき、その情報を用いて

データベース内のテーブルでビューを作成し、検索を行うというアプローチをとっている。さらに、多様なユーザーデータベースに対応するため、データベース接続のために、PHP で利用できるデータベース抽象化レイヤクラスである PDO (PHP Data Object) を使用している。

2.3. 研究者のためのマイ・データベース・システムの開発

及川は、研究者個人が、分析の対象として、もしくは研究の成果をまとめたものとして作られるデータベースの概念を考えている[3]。一般的に、人文系のデータベースは標準化が困難で、データベースの内容は、研究者それぞれの研究内容や成果と深く関連している。したがって、データベースの対象となる研究資料が同じであっても、データベースの内容は研究者ごとに異なったものになりがちで、それぞれの研究者のニーズに応えたデータベースが作られることが必要となる。本来、データベースは、周到的な準備を経て、組織的、計画的に作成され、複数人で共有されるものであるが、マイ・データベース・システムは、研究者個人（場合によっては、グループもあり得るが）のデータベースとして、データベース内の項目の定義を変更したり、再編集したりといった作業を容易に行えるよう設計されている必要がある。また、人文系の研究者でも簡単に作れ、それを研究に活用できるようなシステムにもなっていなければならない。

本研究では、このマイ・データベース・システムの概念を念頭に置いている。

2.4. 地理情報システムを用いた遺跡データベース構築

横山らは、遺跡立地と地形及び自然環境との関連を解明するためのデータベースについて概説し、その構築における問題点などについて述べている[4]。

このデータベースは、青森県、岩手県、秋田県及び宮城県を対象範囲として構築されており、格納されているデータは、各県の教育委員会が編集した遺跡資料や関連する地理情報からなっている。

機能として、遺跡データの管理や水系や標高などと組み合わせた遺跡分布の表示、地理データの操作、検索結果の統計処理などを可能としており、今後もさらなる拡張が考えられている。

3. 対象とするデータ

本システムで扱うデータは、主に遺跡データと遺物データの 2 つに分けられる。以下、これらについて説明する。

3.1. 遺跡データ

遺跡データには、以下のような項目がある。

- ・ 遺跡 ID：ユーザーが定義する遺跡固有の ID
- ・ 遺跡名：遺跡の名称
- ・ 国名：遺跡が属する国名
- ・ 地方名：遺跡が属する地方名
- ・ 都道府県名：遺跡が属する都道府県名
- ・ 支庁市郡名：遺跡が属する支庁市郡名
- ・ 緯度：遺跡の代表点（ユーザーが定義する）における緯度
- ・ 経度：遺跡の代表点における経度
- ・ 標高：遺跡の代表点における標高
- ・ 備考：上記の項目に当てはまらない情報

3.2. 遺物データ

遺物データには、以下のような項目がある。

- ・ 遺跡 ID：遺物が出土した遺跡の ID
- ・ 発掘次数：第何次の発掘で発掘されたかを示す番号
- ・ 発掘年月日：遺物が発掘された年月日
- ・ 遺物番号：他の遺物と区別するために付けられる番号
- ・ 遺物番号枝番：遺物番号を補助する番号。遺物同士に関連がある場合に、遺物番号を同じにし、遺物番号枝番で区別する。
- ・ 大別：製法や原材料など遺物の大まかな分類
- ・ 器種：形状や用途など遺物の種別
- ・ 区画名：遺物が発掘された区画の名前。区画とは、一般的に 3 メートル四方で囲まれた領域のことである。
- ・ 土層名：遺物が発掘された場所の土層の名前。これは、遺物が発掘された場所の深さを示すものである。
- ・ 緯度：遺物が発掘された場所の緯度
- ・ 経度：遺物が発掘された場所の経度
- ・ 標高：遺物が発掘された場所の標高
- ・ 備考：上記の項目に当てはまらない情報

4. 使用するシステム

ここでは、本システムを設計、実装するために利用する既存のシステムについて概説する。

4.1. Ajax

Ajax (Asynchronous Javascript + XML) とは、Web ページのリロードを伴わずに、サーバーと XML 形式 (HTML 形式でも可) のデータのやり取りを処理する技術である。従来の Web アプリケーションでは、サーバーにリクエストを送信後、レスポンスを新たに Web ページとして受け取り、ロードし直さなければならな

かったが、Ajax では、ユーザーの操作や画面描画の裏で、サーバーと非同期に通信することで、サーバーの存在を感じさせないシームレスな Web アプリケーションを実現することができる。

4.2. Adobe AIR

4.2.1. 概要

Adobe AIR (以下 AIR) [5]とは、アドビ・システムズ社が提供するデスクトップアプリケーション/ウィジェットの実行環境である。デスクトップアプリケーションとあるが、もちろんサーバーとの連携も可能である。AIR とは、厳密には実行環境の事だが、AIR で動くアプリケーションそのもの、またはその開発技術を指すこともある。また、クライアントにある他の Java アプリケーションと AIR アプリケーションを仲介し、オブジェクトの受け渡しを可能にする Merapi [6]という API も誕生しており、様々な利用方法が考えられるようになっている。

4.2.2. AIR の特徴

以下に、AIR の特徴や機能の一部を挙げる。

(1) OS の壁を越えて動作可能

ランタイムさえインストールされていれば、アプリケーションを実行するのに OS を問わない。対応 OS は WindowsXP, Vista, MacOS X, Linux 等。

(2) 開発には新しい知識を要しない

HTML/CSS/Javascript (Ajax), Flex/Flash などの既存の Web アプリケーション開発技術での開発が可能である。

(3) サーバー側の開発技術を選ばない

サーバー側は、Java, PHP, Perl, Ruby, .NET 等と、クライアント側同様に様々な技術を選択することが可能である。もちろんサーバー処理のない AIR アプリケーションを開発することも可能である。

(4) ローカルファイルへのアクセスが可能

ユーザーのコンピュータのデスクトップ上にあるファイルへの読み書き等、ローカルリソースともある程度連携することが可能である。

(5) データベースに対応

組み込み式データベースである SQLite [7] に対応している。

4.2.3. AIR アプリケーションの開発と実行

AIR アプリケーションの開発には、Adobe AIR SDK (Software Development Kit) [8] を使用する。AIR API を利用するためのフレームワークやデバッグに必要なコマンドライン等、開発に必要なファイルがいくつか同梱されている。

AIR アプリケーションを開発するには、アプリケーションのコンテンツファイルの他にも、アプリケーション記述ファイルというファイル名やバージョンなど

アプリケーションに関する様々な情報が書かれた XML 形式のファイルが必要である。アプリケーション記述ファイルのテンプレートファイルが SDK に同梱されている。

開発したアプリケーションを実行するためには、SDK に同梱されている adl (AIR Debug Launcher) を用いればよいが、これは名前の通りデバッグに用いるもので、本来はパッケージを行い、パッケージされた物をコンピュータ上にインストール、実行する形になる。パッケージされた物は、AIR さえインストールされていれば、実行することができる。

4.3. Ext JS

Ext JS とは、Ajax アプリケーションを開発するための、javascript フレームワークである[9]。これを用いると、比較的容易にリッチなユーザーインターフェースの構築が可能となる。javascript で動作するので AIR との連携にも向いている。

4.4. Google Maps API

Google Maps API とは、Google Map を使用するための API である[10]。専用ページで Google Maps API キーを取得することで、誰でも自分の Web ページ/Web アプリケーションに Google Map を取り入れることができるようになる。API を利用するには、新しい知識は特に必要なく、ユーザーは単に javascript でインターフェース部分を記述するだけで、容易に Google Map をページに埋め込むことができる。

5. 設計

5.1. システム構成

まず、図 5.1 に本システムの目指す動作モデルを示す。図 5.1 に示すように、本システムは、大きくクライアント側プログラムとサーバー側プログラムの 2 つに分けられる。

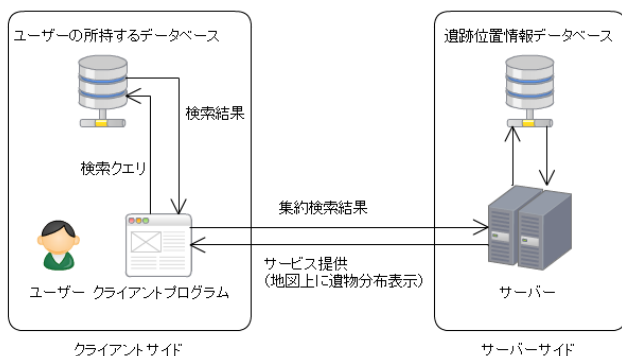


図 5.1 本システムの動作モデル

5.1.1. クライアント側プログラム

クライアント側プログラムでは、ユーザーの所持す

るデータベースへ完全にローカルな環境で接続する。したがって、ユーザーの所持するデータベースの接続に必要な情報を外部へ漏らすことなく、システムの利用を可能にする。さらに、接続したデータベースのどの部分から検索を行うかを指定するために、接続したデータベース内のテーブルを用いて、ユーザー自身がビューを作成できるようにする。ビューとは、1 つ以上のテーブルから任意のデータを選択し、表したものである。ビューを使用すると、複数のテーブルを1つの表であるかのように見ることができる。

そして、作成したビューからデータ検索を行う。本システムにおける検索では、遺物の条件指定の他に、位置に関する検索対象の指定も行えるようにする。設定する項目は、「検索対象全体」と「検索単位」の2つである。検索対象全体とは、遺物の出土分布を表示する範囲であり、検索単位とは、遺物の出土分布を表示する単位である。検索対象全体については、さらに詳細な情報も設定する。例えば、検索対象全体を「都道府県 (京都府)」とし、検索単位を「遺跡」と設定して検索を行うと、京都府の中から遺跡単位で、遺物の出土分布が地図上に表示されるようにする。検索対象全体や検索単位といった集約単位は、図 5.2 に示すような包含的な上下関係を持っており、検索単位として設定できるのは、検索対象全体よりも下の関係にあるもののみとする。例えば、検索対象全体を「都道府県」、検索単位を「地方」とすることはできない。

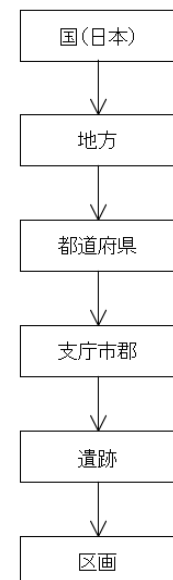


図 5.2 集約単位の上下関係

5.1.2. サーバー側プログラム

サーバー側プログラムでは、クライアントから送信されてきた検索結果をもとに、遺物の出土分布を

Google Map 上に表示することが目的である。遺物の出土分布の表示に必要な情報は、5.1.1 で述べた検索対象全体と検索単位、さらに各検索単位における検索結果の個数である。したがって、クライアントから送信される情報は、検索対象全体、検索対象全体の詳細、検索単位、各遺跡 ID における検索結果数である。各遺跡 ID における検索結果数から、各検索単位における検索結果数を算出する。また、Google Map に遺物の出土分布を表示するには、Google Maps API が必要だが、Google Map を表示するのはサーバー側なので、ユーザーがそれを導入する必要はない。現在、Google Map による遺物分布の表示は、Web ブラウザ（ユーザーのコンピュータ上で、標準ブラウザとして設定されているもの）で表示されるようになっている。

5.2. 情報の管理

遺物の出土分布を地図上に表示する上で必要になってくるのは、各集約単位の位置情報である。これらの位置情報は、全てサーバー側のデータベース上で管理される。

5.2.1. 遺跡の位置情報の管理

遺跡の位置情報は、ユーザーによって異なる可能性が考えられるので、ユーザー固有のテーブルを作成し、そこに遺跡データを保存するようにする。格納される遺跡データは、3.1 で示した通りである。

5.2.2. 遺跡以外の位置情報の管理

国、地方、都道府県、支庁市郡などの遺跡よりも上の関係にある集約単位の位置情報は、システム側で定義し、サーバー側のデータベースに以下の項目を保存する。

- ・ ID：システム側で定義される識別子
- ・ 名前：集約単位の名称
- ・ 国：属する国の ID
- ・ 地方：属する地方の ID
- ・ 都道府県：属する都道府県の ID
- ・ 支庁市郡：属する支庁市郡の ID
- ・ 緯度（代表点）：代表点の緯度。代表点は、分布を表示する際に用いる。
- ・ 経度（代表点）：代表点の経度
- ・ 緯度（中心点）：中心点の緯度。中心点は、地図の初期位置設定の際に用いる。
- ・ 経度（中心点）：中心点の経度
- ・ 拡大率：Google Map の拡大率

「地方」「都道府県」「支庁市郡」の項目に関しては、集約単位の種類によっては、項目が存在しない場合がある。例えば、都道府県の位置情報には、都道府県より下の関係にある支庁市郡の項目は存在しない。

5.3. ユーザー情報の管理

5.2.1 に示した通り、遺跡の位置情報は、ユーザー固

有のテーブルに保存される。したがって、ユーザー別での登録となり、ユーザー認証機能が必要になってくる。その際に必要となるユーザー情報（ユーザー ID とパスワード）も、サーバー側データベースで管理する。

6. 実装

6.1. クライアント側プログラムの実装

6.1.1. インターフェース

本データベースシステムにおけるクライアント側プログラムのインターフェースは、AIR と Ext JS を用いて、HTML+javascript で実現している。

6.1.2. データベースの操作

データベース接続や検索などのデータベースの操作は、AIR API を用いて行っている。AIR API は、SQLite のデータベースファイルに対する処理が可能だが、それ以外のデータベース管理システムには対応していないので、SQLite 以外のデータベース管理システムを扱う場合は、AIR API 以外の方法で、データベースの操作を行う必要がある。

6.1.3. 検索結果のサーバーへの送信

検索結果のサーバーへの送信は、AIR API を用いて行っている。送信方法は、GET メソッドとなっている。送信される変数は、クライアント識別変数 `client`、検索対象全体 `whole`、検索対象全体詳細 `whole_detail`、検索単位 `unit`、遺跡 ID ごとの検索結果数が格納された配列 `rslist` である。クライアント識別変数とは、サーバー側プログラムに、本システムのクライアント側プログラムから結果が送信されていることを示すための変数である。それ以外は、5.1.2 に示した通り、遺物の出土分布表示のために用いる。

6.2. サーバー側プログラムの実装

6.2.1. インターフェース

本データベースシステムにおけるサーバー側プログラムは、データベースを扱うため、HTML+PHP で実現している。サーバー側データベースのデータベース管理システムは MySQL [11] を用いている。

6.2.2. 遺物の出土分布の表示

クライアント側プログラムから送信されてきた変数を受信し、あらかじめサーバー側に用意されている各集約単位の位置情報データベースを用いて、地図上に遺物の出土分布を表示する。

まず、クライアント識別変数でクライアントの識別を行う。クライアントが、本システムのクライアントプログラムでない場合は、検索結果の表示を行わないようにする。次に、検索対象全体と検索対象全体詳細を用いて地図の初期位置を設定する。その際に、5.2.2 で示した中心点の緯度と経度、地図の拡大率を用いる。

そして、各検索単位で検索結果数をまとめ、地図上にマーカーを立てていく。マーカーを設置する場所は、5.2.2 で示した代表点の緯度と経度を用いる。

7. 動作例

ここでは、現在実装済みの部分の動作例を示す。

7.1. システムの起動

システムを起動すると、図 7.1 に示すようなウィンドウが表示される。ウィンドウ内には、タブパネルが表示されており、「DB 接続設定」「DB 検索」「マニュアル（未実装）」の 3 つのタブが用意されている。DB 接続設定のタブにて、ユーザーの所持するデータベースへ接続を行う。最初、DB 検索のタブは、データベースの接続設定が完了しないと利用できないようになっている。また、ウィンドウ右上の「status:」の部分で、システム起動時にインターネットに接続されているかどうかを確認することができる。表示される値は、接続されているなら「online」、接続されていないなら「offline」となっている。



図 7.1 システムの初期ウィンドウ

7.2. データベースへの接続

最初、データベース接続設定のタブには、図 7.2 に示すようなパネルが表示されている。このパネルでは、接続するデータベースの指定を行う。図 7.2 で示されるようなセレクトボックス形式で、データベースの管理方法を選択する。選択できる項目として「SQLite」「MySQL」「PostgreSQL」「Excel (csv 形式)」「Access (mdb 形式)」があるが、現在対応しているのは SQLite のみである。

次に、図 7.3 に示すように、データベース接続に必要な情報をフォームに入力する。例えば、SQLite の場合は、データベースファイルへのパスのみである。フォームに入力された情報を用いて、データベースへ接続する。接続に成功すると、図 7.4 に示すようなパネ

ルが表示され、接続に失敗した場合は、エラー内容がパネルに出力される。

このパネルでは、接続したデータベースのスキーマ情報を確認できる。確認できる情報は、データベース名、そのデータベース内の全テーブル名、各テーブルの全カラム名とデータ型である。図 7.4 の場合、“rdbtest.db”がデータベース名、その下にある“testdat”がテーブル名、さらにテーブル名をクリックすると、カラム名とデータ型の一覧が表示されるようになっている。



図 7.2 データベース管理方法の選択

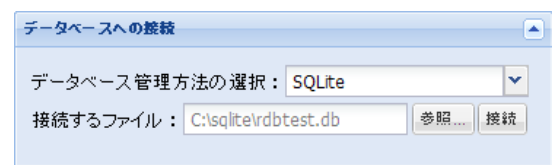


図 7.3 データベース接続に必要な情報の入力

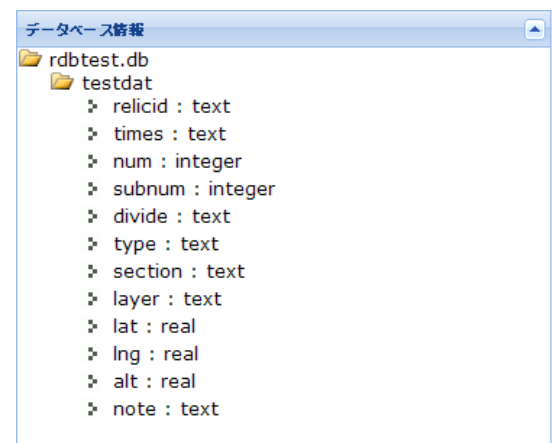


図 7.4 データベースのスキーマ情報の確認

7.3. ビューの作成

データベースへの接続が完了すると、次に、データベース検索を行うために、ビューの作成を行う。図 7.5 に示すパネルで、3.2 で示した各項目のデータが、接続されたデータベース内のどれにあたるかを指定する。左のセレクトボックスでテーブル名、右のセレクトボックスでカラム名を指定し、セレクトボックスの値は、データベース内にあるものが自動でロードされる。カラム名は、テーブル名が決まらないと選択できない。

また、遺物番号枝番の項目は、遺物番号のカラムを登録しないと、登録できないようになっている。

結合条件とビュー名については、現在未実装なので、入力できないようになっている。結合条件は、制限なしに設定できるようにすることを考えているので、インターフェースを変更する可能性がある。ビュー名は、現在固定となっており、“RelicDB”というビューが作成される。既に同じ名前のビューが存在する場合、前にあったものが削除され、新たに作成されるようになっている。

図 7.5 カラム情報の登録

7.4. データの検索

まず、図 7.6 に示すパネルで、位置に関する検索対象の指定を行う。検索対象を指定せずに、データの検索をすることは可能だが、遺物の出土分布を表示するには、検索対象を指定する必要がある。パネルには、検索対象全体、検索対象全体詳細、検索単位を指定する3つのフォームがあり、検索対象全体詳細や検索単位を指定するには、検索対象全体を指定しなければならない。検索対象全体詳細は、検索対象全体を決定すると、画面上に図 7.7 のようなウインドウが表示され、ここで決定する。選択した検索対象全体によって、表示されるウインドウが異なっている。図 7.7 において、左から「地方」「都道府県」「支庁市郡」の詳細を決定するウインドウである。目的の値を探しやすいように、自身より上の関係にある集約単位(この場合は「地方」)から順次決定していくようになっている。検索対象全体として「国」を選択した場合は、自動的に「日本」が選択される。また、検索単位は、選択した検索対象全体よりも下の関係にあるものがセレクトボックスにロードされ、ここから選択する。

図 7.6 検索対象の指定

図 7.7 検索対象全体 決定ウインドウ

次に、図 7.8 に示すパネルで、遺物の検索条件の指定を行う。検索条件の指定は、ビュー作成時に登録を行ったカラムのみからできるようになっている。検索は、基本的にセレクトボックス形式で行い、セレクトボックスの値は、作成したビューのカラム内にあるものの全てがロードされる。遺物番号／遺物番号枝番の項目は、セレクトボックスによる番号直接指定か、テキストフィールドによる番号範囲指定かを選択できる。また、備考の項目は、テキストフィールドによる文字列部分検索となっている。

検索結果は、図 7.9 に示すように、表形式で表示される。パネル内のインターフェースを操作することによって、検索結果のソートなども行える。

図 7.8 検索条件の指定

検索結果：15件のデータが見つかりました

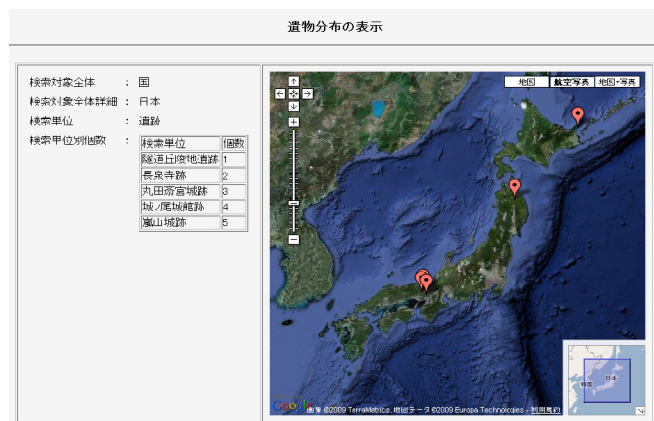
遺跡ID	発掘回数	遺物番号	遺物番号付番	大別	器種	区画	土層	備考
ID1	次数1	1	0	大別1	器種1	区画1	土層1	備考1
ID2	次数2	2	1	大別2	器種2	区画2	土層2	備考2
ID2	次数2	2	1	大別2	器種2	区画2	土層2	備考2
ID3	次数3	3	1	大別3	器種3	区画3	土層3	備考3
ID3	次数3	3	1	大別3	器種3	区画3	土層3	備考3
ID4	次数4	4	1	大別4	器種4	区画4	土層4	備考4
ID4	次数4	4	1	大別4	器種4	区画4	土層4	備考4
ID4	次数4	4	1	大別4	器種4	区画4	土層4	備考4
ID5	次数5	5	1	大別5	器種5	区画5	土層5	備考5
ID5	次数5	5	1	大別5	器種5	区画5	土層5	備考5
ID5	次数5	5	1	大別5	器種5	区画5	土層5	備考5
ID5	次数5	5	1	大別5	器種5	区画5	土層5	備考5
ID5	次数5	5	1	大別5	器種5	区画5	土層5	備考5

地図上に分布を表示(Webブラウザで開きます)

図 7.9 検索結果の表示

7.5. 遺物分布の表示

検索結果表示パネル下部に遺物の出土分布表示ボタンがあり、これを押下すると、Web ブラウザ（ユーザーがコンピュータ上で標準ブラウザとして設定しているもの）に専用ページが表示される。サーバーが起動していない、もしくはインターネットに接続されていない場合は、ブラウザ側でページが表示されない旨のエラーが表示される。サーバーへの接続が成功すると、図 7.10 のようなページが Web ブラウザに表示される。画面左に検索対象についての情報が表示され、画面右に GoogleMap が表示される。GoogleMap 上には、検索単位ごとにマーカーが立てられ、これらをクリックすると、その検索単位の ID、名前、検索結果数を参照することができる。なお、この例では、図 7.10 の 5 つの遺跡には、遺跡 ID が「ID1」～「ID5」として登録されている。



※使用している画像の著作権はすべて Google 社にあります

図 7.10 遺物の出土分布の表示

8. おわりに

クライアント側でユーザーの所持するデータベースに接続することで、それをサーバー側にコピーすることなく、考古学データの検索、また地図上に検索結果の分布表示を可能にした。

システムの基幹部分の実装は行ったが、実際にユーザーに利用してもらえるほど十分な機能が実装されていない。ビュー作成時に結合ができない、SQLite 以外のデータベース管理システムを利用できないといった点である。今後の課題として、まだ実装していない機能の実装、既の実装されている機能の改良などが挙げられる。また、実装したシステムを実際に考古学研究者に使用してもらい、性能や使い勝手の評価を行うことも今後の課題である。

参 考 文 献

- [1] 山内祥裕, 宝珍輝尚: “地図 API を用いた遺跡データベースシステムの設計と実装”, 情報処理学会研究報告 2007-CH-74(11), vol.2007, no.49, pp.81-88 (2007).
- [2] Teruhisa Hochin, Fumiaki Kobayashi, Hiroki Nomiya: “Seamless Usage of User’s Databases in Archaeological Database System”, Proceedings of CIPA2009, PS1-13 (2009).
- [3] 及川昭文: “研究者のためのマイ・データベース・システムの開発”, 第 13 回公開シンポジウム「人文科学とデータベース」, pp.25-34 (2007).
- [4] 横山隆三, 千葉史: 地理情報システムを用いた遺跡データベースシステム構築, 情報考古学, Vol.3, No.2, pp.29-39 (1997)
- [5] Adobe AIR: <http://www.adobe.com/jp/products/air/>
- [6] Merapi: <http://merapiproject.net/>
- [7] SQLite: <http://www.sqlite.org/>
- [8] Adobe AIR SDK: <http://www.adobe.com/products/air/tools/sdk/>
- [9] Ext JS: <http://www.extjs.com/>
- [10] Google Maps API: <http://code.google.com/intl/ja/apis/maps/>
- [11] MySQL: <http://www.mysql.com/>