

## 考古学のためのデータベースシステム

## A Database System for Archeology

宝珍 輝尚\*, 中田 充†, 白井 治彦‡, 都司 達夫\*

Teruhisa HOCHIN\*, Mitsuru NAKATA†, Haruhiko SHIRAI‡, Tatsuo TSUJI\*

\* 福井大学 工学部 情報工学科

† 福井大学大学院 工学研究科

‡ 福井大学 工学部

〒 910 福井市文京 3 丁目 9 - 1

\* Department of Information Science, Faculty of Engineering, Fukui University

† Graduate School of Engineering, Fukui University

‡ Faculty of Engineering, Fukui University

3-9-1, Bunkyo, Fukui-shi, Fukui 910 Japan

あらまし: 考古学では, 未整理状態の遺物データ等を管理すること, 整理・分類により知見を得ること, 研究者間で知見を共有すること, ならびに, 分かりやすいユーザインタフェースが求められている。これらはデータベースにとって大きな課題である。本論文では, これらの課題を解決するための, 基本データベース, 導出データベース, ならびに, グラフィカルなユーザインタフェースについて述べる。基本データベースには未整理データをとりあえず格納できる。導出データベースは集合により柔軟性を高めたデータベースであり, データ主導のデータ定義情報を持つ。ユーザインタフェースでは, 地図上での遺物表示が可能である。また, 一乗谷朝倉氏遺跡の遺物を対象とした適用例を示す。適用例について述べる。

**Summary:** Managing un-arranged relic data, obtaining information through arrangement and classification, and sharing the information among researchers, as well as user-friendly interface are required in the archeological research.

These are the serious issues for databases. For the purpose of addressing these issues, this paper describes a basic database, a derived database, and the user interface. Un-arranged data can be stored in a basic database. A derived database is flexible because it is based on sets. This database can be used in a top-down manner through the data-driven information on data stored. Query results can be displayed on a map by using the user interface. Application of this system to Asakura Family Castle Site is also presented.

**キーワード:** 陶磁器データ管理, 柔軟なデータベース, データモデル, グラフィカルユーザインタフェース

**Keywords:** Ceramics Data Management, Flexible Database, Data Model, Graphical User Interface

## 1 はじめに

近年のコンピュータの進歩は目覚ましく、考古学へのコンピュータの導入も盛んに検討されてきている [1, 2]. この中でも、考古学データのデータベース化は多くの研究者から必要とされている。ここでは、様々な個別データベース群とそれらの統合データベースが構築・整備され、必要な情報が迅速に得られることが期待されている。しかし、データベースの構築にあたっては、情報量とデータベースへのデータ投入負荷を考慮することやデータ格納に先立つデータ定義を十分検討することが重要であると指摘され、安易なデータベース構築に警鐘が与えられている [3, 4]. これは、従来のデータベースの欠点とも考えられる。すなわち、データベース構築に先立って全てを決定しておかなければならないということであり、トップダウンにしかデータベースを構築できないということである。考古学研究のように、様々な研究が積み重なって体系をなしてゆく場合は、データベースをボトムアップに構築できる方が望ましいが、従来のデータベースではこれは不可能である。

そこで、本論文では、個別データベースを柔軟に構築でき、しかも、あとから柔軟に統合して統合データベースを構築できるようなデータベースの実現を目的として、ボトムアップにデータベースを構築可能とするデータベースの枠組みについて述べる。本枠組みでは、2種類のデータベースを考える。未整理データを格納するデータベース（基本データベース）と基本データベース中のデータを研究の進展に応じて徐々に整理することを可能とするデータベース（導出データベース）である。基本データベースは、あらゆるデータを格納するため、ならびに、格納したデータの柔軟で効率良い操作を可能とするために、全く構造を持たない非構造化データベースではなく、操作性を考慮した構造化データベースとする。導出データベースでは、物事を集合としてとらえることを従来のデータベースよりもより一層徹底する。すなわち、従来  $n$  項組で表された物事を集合としてとらえる。物事を集合として表現し、集合演算を用いて処理を行う。これにより、データベースの柔

軟性を高めることができる。さらに、従来のデータ定義に相当しデータ主導で変化するシェイプという概念を導入する。これにより、従来のデータベースが持つ、データ定義に基づく問い合わせやトップダウン的なデータ整理といった利点を継承する。

以下、2. で基本データベースについて述べ、3. で、導出データベースについて述べる。そして、4. で本データベースシステムの適用事例について述べる。ここでは、地図上で検索結果を表示するユーザインタフェースについても述べる。最後に、5. でまとめを行う。

## 2 基本データベース

基本DBには、管理したい対象物から人間が抽出したデータ（対象物の重さ、サイズ、考古学データベースであれば発掘日や、発掘場所など）や、対象物から得られる画像・音声・動画といった、いわゆる、マルチメディアデータが未整理の状態に格納される。それらのデータ（測定データ）は、一見、普遍的なデータであるようにみえるが、例えば重さ、大きさなどは測定した人間やその時の環境によって変化する可能性がある。そこで、測定データにはそのデータを得た環境や測定条件などを付加しておく。この測定データと測定条件の組を基本データと呼ぶ。さらに、同じ対象物から得られた基本データは一つの識別子で参照できるようにする。この識別子で識別されるデータを基本エレメントと呼ぶ。基本エレメントには複数の基本データを格納できる。ユーザは基本エレメントから測定条件をもとに所望するデータを決定する。基本DBは基本エレメントの集合である。

基本エレメントの例を図1に示す。図1の基本エレメントの識別子は「12345」である。この基本エレメントには2個以上の基本データが格納されている。測定データは文字列で格納されている。ここでは、便宜的に「,」で区切ってデータを格納する例を示しているが、どのような形式でデータを格納しても構わない。また、この文字列中には「表の写真」と「裏の写真」という文字列を格納している。これが、「表の写真」や「裏の写真」と

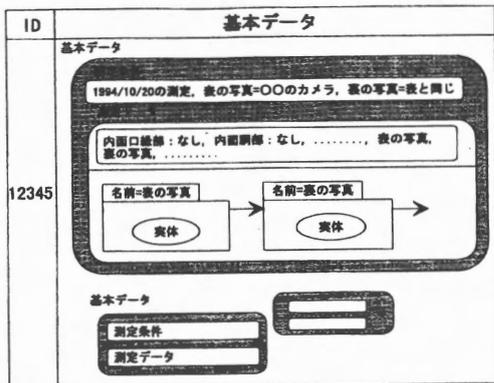


図 1: 基本エレメント

いう名前の付いたマルチメディアデータを一緒に格納していることを表す方法である。「表の写真」や「裏の写真」の有無は、その文字列をサーチすることで確認できる。また、測定条件も文字列として格納されている。

### 3 導出データベース

#### 3.1 データエレメント

データエレメントは導出DBにおいてデータの実体を格納する要素であり、導出DBにおけるデータの取り扱いの最小単位である。導出DBはデータエレメントの集合である。

データエレメントには識別子が付与される。データエレメントには、実データエレメントと仮想データエレメントがある。実データエレメントは、基本DB中の基本データを指す指示エレメントまたはデータ値、ならびに、単位で構成されるデータセルとデータの種類とどのようにデータエレメントが得られたかを表す導出過程で表現される。ここで、データの種類は、length (長さ), weight (重さ), image (静止画), video (動画), string (文字列), numeric (数値) などがあり、ユーザが独自に定義することもできる。

識別子	データセル	種類	導出過程
123	kg 50.5	weight	.....

124	cm	指示エレメント	length	.....
-----	----	---------	--------	-------

単位 データ

図 2: 実データエレメント

基本データ (文字列)

ID=40, 種類=茶碗, 幅=13.32cm, . . . .

データエレメント=(識別子, (cm, 指示エレメント), length, 導出過程)

図 3: 指示エレメント

実データエレメントの例を図2に示す。図には2つの実データエレメントが存在する。上の実データエレメントには「50.5」というデータ値が格納されている。また、単位は「kg」で重さ (weight) である。下の実データエレメントにはデータ値ではなく指示エレメントが格納されている。これは、基本データ中のデータを参照するものである。指示エレメントの概念図を図3に示す。この例では、基本データ中の「13.32」という文字列を指示エレメントで指している。

仮想データエレメントとは、実体を持たず、他のデータエレメントを参照するデータエレメントである。仮想データエレメントは主にデータエレメントを他の導出DBから参照する場合に用いる。

#### 3.2 名前付きエレメント

基本的には、データエレメントに名前を付けたものが名前付きエレメントである。データエレメ

ントの集合、配列、順序集合、ならびに、オブジェクトにも名前を付けて名前付きエレメントとすることができる。オブジェクトに名前をつけて名前付きエレメントとした場合は、オブジェクトが他のオブジェクトから構成されている、いわゆる、複合オブジェクト [8] を構成することになる。名前付きエレメントは、後述のオブジェクトの主要な構成要素であり、名前付きエレメントは必ず一つのオブジェクトに属さなければならない。これに対して、データエレメントは必ずしも名前付きエレメントに対応しているとは限らず、また複数の名前付きエレメントに対応する場合もある。

### 3.3 オブジェクト

オブジェクトは研究者の見方や仮定を反映したデータの単位である。オブジェクトには識別子が付与され、その識別子は導出DB内で一意である。オブジェクトは基本的には名前付きエレメントの集合である。名前付きエレメントの集合は、導出DB中のデータエレメントの部分集合にマッピングされる。オブジェクトは、後述する複数のバンドルに同時に属することができる。また、どのバンドルにも属さないで存在することもできる。

オブジェクトの例を図4に示す。図4には2つのオブジェクトが示されている。いずれも、「内面紋様」、「半径」、「重さ」という名前の名前付きエレメントを持っている。

オブジェクトが自身の識別子の他に、他のオブジェクトの識別子とそのオブジェクトが属する導出DBの識別子から構成されている場合、そのオブジェクトは実体を持たずに他のオブジェクトへの参照ポインタをもつ仮想オブジェクトである。仮想オブジェクトは主にオブジェクトを他の導出DBから参照する場合に用いる。

オブジェクトは、空のオブジェクトに1つ以上名前付きエレメントを挿入することで作成できる。また、既存の2つのオブジェクトに集合演算を施すことにより新たなオブジェクトを作成することもできる。

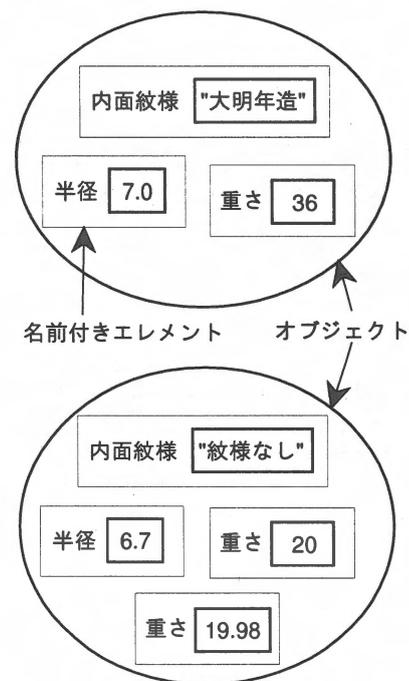


図4: オブジェクト

### 3.4 視点

視点とは、オブジェクトの構成要素である名前付きエレメントの集合の部分集合である。視点を通してオブジェクトをみると、オブジェクトに属する名前付きエレメントのうち、その視点に属している名前付きエレメントのみが見える。部分集合を構成する名前付きエレメントはユーザによって指定される。

視点の例を図5に示す。この例では、2つの視点が設定されている。視点Aから見ると、このオブジェクトは「紋様」、「口径」、「縦」、「横」、「種類」という名前付きエレメントを持ち、視点Bから見ると、このオブジェクトは「紋様」、「半径」、「高さ」、「幅」、「種類」という名前付きエレメントを持つ。

### 3.5 バンドル

バンドルはオブジェクトの集合である。バンドルにも識別子が付与される。バンドルには、ユー

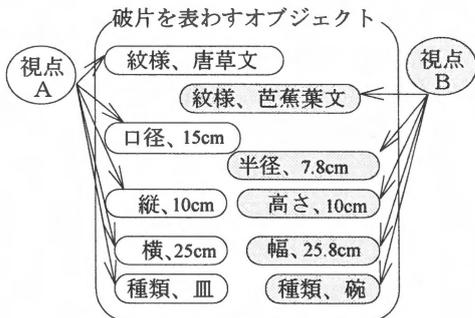


図 5: 視点

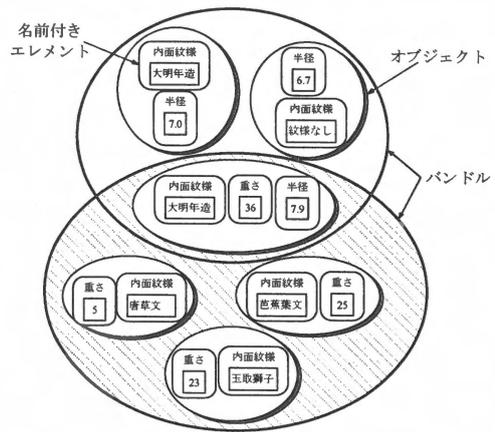


図 6: バンドル

ザが指定した束縛条件を満足するオブジェクトが所属する。束縛条件は、オブジェクトがバンドルに属するために必要な条件である。ここでは、束縛条件を、バンドル条件と複数のオブジェクト条件を論理和で結合したものとする。バンドル条件は、例えば、「高さを持つもの」といった、ユーザがその見方や仮定のもとに指定した条件である。オブジェクト条件はバンドル条件は満たさないがバンドルに属するオブジェクトを表わす条件であり、一つのオブジェクト条件は”OID=オブジェクト識別子”という形で表される。ここでOIDはオブジェクトの識別子を表わす予約語である。オブジェクト条件を導入した理由は、(1) どのような見方や仮定にも例外が存在すること、(2) 必ずしも的確な条件を指定できるとは限らないことである。オブジェクト条件により、バンドル条件を満たしていないオブジェクトをバンドルに無条件に属させることができる。これにより、とりあえずバンドルを生成し、ある程度オブジェクトの数がそろった段階で、それらのオブジェクトの共通性や性質を考慮した上で改めて正確なバンドル条件を定義することが可能になる。

バンドルの例を図6に示す。この例には、2つ

のバンドルがある。上のバンドルには「半径」という名前付きエレメントを持つオブジェクトが属しており、下のバンドルには「重さ」という名前付きエレメントを持つオブジェクトが属している。「半径」と「重さ」の両方の名前付きエレメントを持つオブジェクトは両方のバンドルに属する。

バンドルはバンドル条件を指定して定義することで作成される。バンドル条件を満足するオブジェクトはそのバンドルに属することが可能となる。また、前述のように、バンドル条件を満足しないオブジェクトをそのバンドルに属させることも可能である。さらに、既存のバンドルに集合演算を施すことにより新たなバンドルを作成することもできる。

### 3.6 シェイプ

従来のデータベースにはスキーマが存在する。これは、データベースの形を表す情報であり、データ格納に先立って決定されていることが基本である。従来のデータベースではこのスキーマをもとにデータの問い合わせを行うことができる。また、スキーマが存在することで、トップダウン的

にデータを整理してゆくことができる。

ここでは、導出DBがどのようなデータベースエレメントから構成されているかを表す情報としてシェイプを考える。

シェイプは、基本的には、名前付きエレメントの名前 *name* とその名前を同じくする名前付きエレメントのデータ型の集合 *DT* の組 (*name*, *DT*) の集まりである [19]。組 (*name*, *DT*) をシェイプエントリと呼ぶ。オブジェクト、視点、ならびに、バンドルは、それぞれ、名前付きエレメントの集合であるともみなせる。そこで、それらに関するシェイプを、それぞれ、オブジェクトのシェイプ、視点のシェイプ、バンドルのシェイプと呼ぶ。オブジェクトのシェイプは、オブジェクト中の名前付きエレメントに対するシェイプエントリの集合である。視点のシェイプは、視点中の名前付きエレメントに対するシェイプエントリの集合を *S* とすると、組 (視点名, *S*) で表される。同様に、バンドルのシェイプは、バンドル中の名前付きエレメントに対するシェイプエントリの集合を *S* とすると、組 (バンドル名, *S*) で表される。さらに、導出データベース中の全てのオブジェクトに対しても同様のシェイプが考えられる。これは、視点とバンドルに関しても同じである。導出データベース中の全てのオブジェクト、視点、バンドルに対するシェイプを、おのおの、 $S_{obj}(db)$ ,  $S_{persp}(db)$ ,  $S_{bndl}(db)$  とすると、導出データベースのシェイプは、3つ組 ( $S_{obj}(db)$ ,  $S_{persp}(db)$ ,  $S_{bndl}(db)$ ) で表される。

図4の右側のオブジェクトに対するシェイプの例を図7に示す。このオブジェクトは、「重さ」という名前の名前付きエレメントを2つ持ち、それらの値のデータ型が「int」と「float」である。従って、名前が「重さ」のエントリは、データ型の情報として「int」と「float」を持つ。

次に、シェイプの振舞いについて述べる。シェイプは次の2つの条件を満足する。

**条件1** 全ての名前付きエレメントに対して、その名前とデータ型に対応するエントリがシェイプ中に存在しなければならない。

**条件2** どの名前付きエレメントにも存在しない名前はシェイプ中のエントリに存在してはな

(内面紋様, { string })  
 (半径, { float })  
 (重さ, { int, float })

図7: オブジェクトのシェイプ

らない。また、ある名前 *A* を持つ全ての名前付きエレメントのデータ型として存在しないデータ型が、シェイプ中のその名前 *A* を持つエントリ中に存在してはならない。

条件1は、現在のシェイプに存在しない名前やデータ型を持つ名前付きエレメントが挿入されると、その名前やデータ型が含まれるようにシェイプが変わらなければならないことを意味している。この場合、名前付きエレメントが挿入されたオブジェクトのシェイプが変わり、これに伴い、視点やバンドルのシェイプにも変更が必要な場合にはそれらも変化する。条件2は、名前付きエレメントの削除によりその名前付きエレメントの名前やデータ型を持つような名前付きエレメントが存在しなくなった場合に、シェイプからその名前やデータ型を削除しなければならないことを意味している。

シェイプはデータベースの形を表わすという点ではスキーマと共通点があるが、データ格納に先立って決定している必要がない、データベース中のデータに依存するという点でスキーマとは異なる。シェイプを持つことで、従来のデータベースの利点である、データの問い合わせやトップダウン的なデータの整理を可能とすることができる。

## 4 適用事例

### 4.1 遺物データ

本論文では、遺物データとして一乗谷朝倉氏遺跡から出土した茶碗と皿の破片を対象とする。

出土した破片には、識別番号が割り当てられて保管されている。また、明らかに複数の破片が互

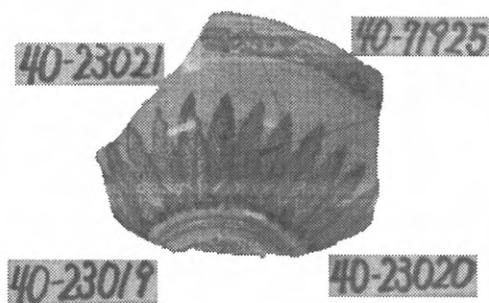


図 8: 破片の例 (1)

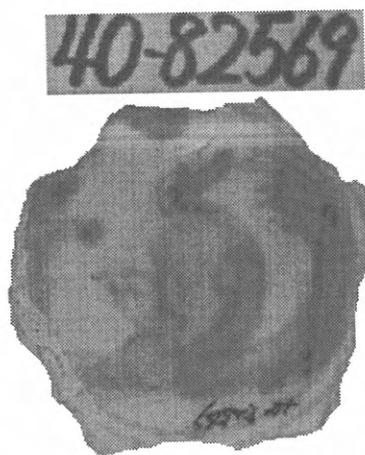


図 9: 破片の例 (2)

いに合致すると認められる場合には、接合され一体型破片として保管されている。

図 8 と図 9 に対象とした破片を示す。図 8 は碗の外側の部分であり、この破片からは口、胴、腰の各部分にどのような模様が入っているか、口は内と外のどちらに沿っているかといった情報が得られる。図 8 の場合、口部分は四方嚮という模様であり、胴部分は芭蕉葉文という模様、腰部分には界線と呼ばれる線が二本入っている。また、口は外側に反った端反りである。さらに、図 1 の破片は 4 つの破片から構成される一体型破片である。図 9 は皿の内側の底部分であるが、ここにどのような模様が入っているかという情報が得られる。図 9 の場合、鳥の絵である。他にも、底の形状が高台かごけ底か、口半径、高台高さ、高台半径、

全体高さといった大きさに関する情報が得られる。

## 4.2 プロトタイプシステム

ワークステーション HP 9000S730 (HP-UX 9.0) 上で、商用のオブジェクト指向データベース管理システム UniSQL<sup>1</sup> を使用して C++ 言語によりプロトタイプシステムを構築した。

基本 DB では、UniSQL のデータベース中に基本エレメントを格納するためのクラスを定義し、さらに、科学データ一般に対応できるようなクラス階層としている [7, 14]。現在、約 1000 件の破片データを基本 DB 中に格納している。破片データとして、碗か皿かの種別、口縁部、胴部、腰部、底の内外部の紋様、全体の高さ、高台の高さ、半径、高台の半径、口の形状、底の形状、その他の特徴を格納している。

基本 DB 中の基本データを導出 DB から参照するための枠組みである指示エレメントは、流動的データ型 [16, 15] を用いて実現されている。流動的データ型とは、バイト列で表現されているマルチメディアデータの不連続領域の操作を可能とするためのデータ型である。流動的データ型においては、不連続領域のバイト列を仮想セグメントと呼び、バイト列全体を実データと呼ぶ。仮想セグメントから実データの一部への写像は指示エントリと呼ばれるデータ構造体を用いて表現される。指示エントリは導出手続きと呼ばれるユーザ定義の関数により実データから生成される。また、仮想セグメントは、対応関数により動的に作成される。図 10 は基本 DB 中のデータの一部を導出 DB 中のデータとして参照している例である。図 10 の基本 DB 中の複数の破片のイメージデータのうち、真ん中の一つの破片のみを導出 DB のデータとして参照している。上下二つの画像の対応をとっているのが指示エントリである。このように、流動的データ型の機構により、基本データの一部を仮想データとして参照することが可能である。また、流動的データ型にはデータ型変換の機能があり、文字列データとして格納されているデータを導出データベースにおいて数値データとして参

<sup>1</sup>UniSQL は UniSQL 社の登録商標である。

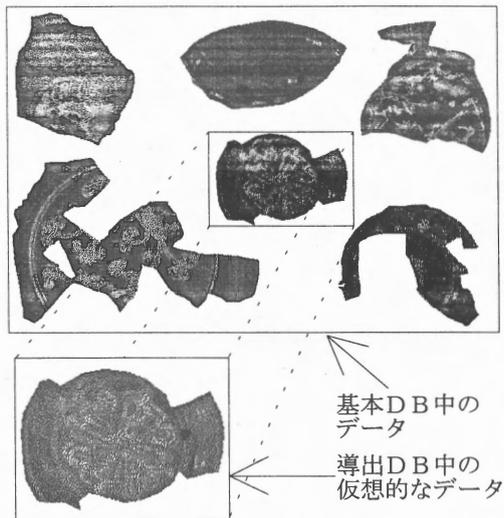


図 10: 仮想セグメントの例

照するといったことが可能である。

導出DBを管理するシステムは、基礎部分の設計と実装が終了している[12]。データエレメント、名前付きエレメント、オブジェクトの要素をUniSQLのデータベース中に定義し、それらを用いたデータの格納が可能である。また、データの削除、検索も可能である。ただし、これらの操作は、現在、C++言語の関数により実行する必要がある。

#### 4.3 ユーザインタフェース

ユーザインタフェースは、将来情報をネットワーク公開することも考慮して、HTML文書として作成している。本インタフェースでは、地図上で出土品の分布をグラフィカルに表示できるようにしている。実行例を図11に示す。図11では、土師の皿の出土分布を求めるように指定している。この検索結果を図12に示す。発掘場所を表すグリッドごとに地図上で破片の出土数に応じて色が変わるようになっており、一目で出土数の多いところや出土範囲が分かる。

本インタフェースは、現在、前述のプロトタイ

プシステムと独立に動作している。検索対象のデータは第50次発掘調査で得られたデータで、発掘場所(グリッド)、地層、大別、器種、備考という項目がある。

本インタフェースでは、データの検索ならびに検索結果の表示に、CGI機能[20]を使用している。現在、データの検索にはPerlを用いている。データの表示にはPerlならびにC言語を用いており、検索結果に応じて地図上の変更箇所を書き換えを行っている。グリッドは、地図上で横方向I~V(14本)、縦方向60~80(21本)の記号で表した線を用いて等間隔に分割しており、グリッドはその記号(例えば、R70)を指定することによって任意に決められる。

## 5 おわりに

本論文では、遺物データを柔軟に管理するデータベースの枠組みについて述べた。この枠組みでは、従来のデータベースでのようにはじめデータ定義を行う必要がない。遺物から得られたデータを、データ定義を行わずに、とりあえずデータベースに格納することができ、後からデータの整理を行うことができる。この枠組みの主な特徴は、(1)基本DBと導出DBという2種のDBを導入したこと、(2)物事を徹底して集合ととらえたことである。このプロトタイプシステムを構築し、また、地図上で検索結果を表示できるようにしている。

今後は、基本DBと導出DBのグラフィカルユーザインタフェースの設計と実現、本論文で述べたユーザインタフェースとプロトタイプシステムの連動、データの導出過程の管理の実現、大量の遺物データ管理への適用が課題である。

## 謝辞

データの収集、分類作業、ならびに、考古学データの管理に関する議論などでお世話になっている福井県立一乗谷朝倉氏遺跡資料館の岩田隆氏、水村伸行氏に深謝いたします。

◆◆ 出土箇所の出土品個数別色分け ◆◆

1. イコウ位置

- 土壘下盛土イコウ
- 土壘内側イコウ
- 青灰色土田イコウ
- 黒灰色土田イコウ

○ 直接出土位置を入力するにはこちらからどうぞ

2. 大別<出土品の作成法（出身）、原料>

- 越
- 土師
- 染付
- 中国

○ 直接大別を入力するにはこちらからどうぞ:

3. 器種<出土品の形状・種類>

- 壺
- 樽
- 皿
- 碗

○ 直接器種を入力するにはこちらからどうぞ:

データ選択完了 入力やり直し

図 11: 検索指定の例



イコウ:.\* 大別:土師 器種:皿

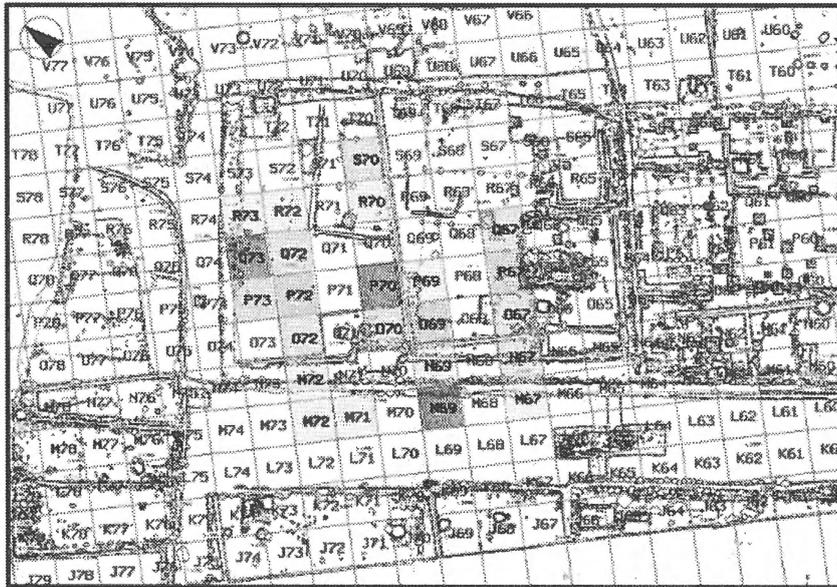


図 12: 検索結果の例

## 参考文献

- [1] 特集・考古学とコンピュータ, 考古学ジャーナル, No. 215, pp.2-39 (1983).
- [2] 特集・考古学とコンピュータII, 考古学ジャーナル, No. 294, pp.4-25 (1988).
- [3] 及川 昭文: 考古学データベースとその課題, 考古学ジャーナル, No. 215, pp.15-20 (1983).
- [4] 及川 昭文: 情報化社会の考古学, 考古学ジャーナル, No. 294, pp.15-20 (1988).
- [5] 上島 紳一, 大月 一弘, 森下 淳也, 田中 克己: 歴史的資料を対象としたサイエンティフィックデータベースのシステム設計, 電子情報通信学会技術報告, DE93-47 (1993).
- [6] 打浪 清一: フィールド調査データ処理におけるマルチメディアデータベース, 情報処理, Vol.28, No.6, pp.773-783 (1987).
- [7] 大平 晃洋, 中田 充, 宝珍 輝尚, 都司 達夫: 考古学データベースの一構成法, 平成6年度電気関係学会北陸支部連合大会, E-45, pp.339 (1994).
- [8] 加藤和彦: オブジェクト指向データベースシステムの記憶構造, 情報処理, Vol.32, No.5, pp.532-539 (1994).
- [9] 杉田 繁治: 人文科学におけるマルチメディアデータベース, 情報処理, Vol.28, No.6, pp.765-772 (1987).
- [10] 中田 充, 宝珍 輝尚, 都司 達夫: サイエンティフィックデータベースのためのデータモデルの一提案, 情報処理学会研究報告, データベースシステム研究会 101-9, pp. 65-72 (1994).
- [11] 中田 充, 宝珍 輝尚, 都司 達夫: サイエンティフィックデータベースにおけるデータモデルの考察, 平成6年度電気関係学会北陸支部連合大会, E-44, pp.338 (1994).
- [12] 中田 充, 宝珍 輝尚, 都司 達夫: サイエンティフィックデータベース管理システム DREAM の設計, 情報処理学会平成7年後期全国大会, 5E-09 (1995).
- [13] 中田 充, 宝珍 輝尚, 都司 達夫: サイエンティフィックデータモデルの一評価, 電子情報通信学会 データ工学研究会 DE95-64, 信学技報 Vol. 95, No. 287, pp. 113-120 (1995).
- [14] 中田 充, 宝珍 輝尚, 都司 達夫: サイエンティフィックデータベースのための一次データの一管理法, 情報処理学会平成8年前期全国大会, 2Q-2 (1996).
- [15] 原田 正則, 宝珍 輝尚, 中田 充, 都司 達夫: 流動的データ型を用いたマルチメディアデータ参照機構の設計と実装, 1996年度電子情報通信学会総合大会, D-64. (1996)
- [16] Hochin, T. and Tsuji, T.: On the Application Interface of the LIQUID data type for flexible manipulation of multimedia data, Proc. of Int'l Symposium on Advanced Database Technologies, pp. 200-206 (1994).
- [17] 中田 充, 宝珍 輝尚, 都司 達夫: 考古学データの柔軟な管理をめざしたデータベースシステムの設計と実装, 日本情報考古学, Vol. 1, No. 1, pp.46-54 (1996).
- [18] 宝珍 輝尚, 安達 政伸, 原田 正則, 中田 充, 都司 達夫: 考古学のための一次データベースについて, 日本情報考古学会 第1回大会, pp. 62-67 (1996).
- [19] 宝珍 輝尚, 中田 充, 都司 達夫: インスタンスベースのデータベースにおける柔軟なスキーマについて, 情報学第53回全国大会, 3R-6, pp. 3-29 - 3-30 (1996).
- [20] 藪 暁彦, 田辺 茂也: HTML早わかり マイ・ホームページを作ろう, インターナショナルトムソン・パブリッシング ジャパン (1995).