

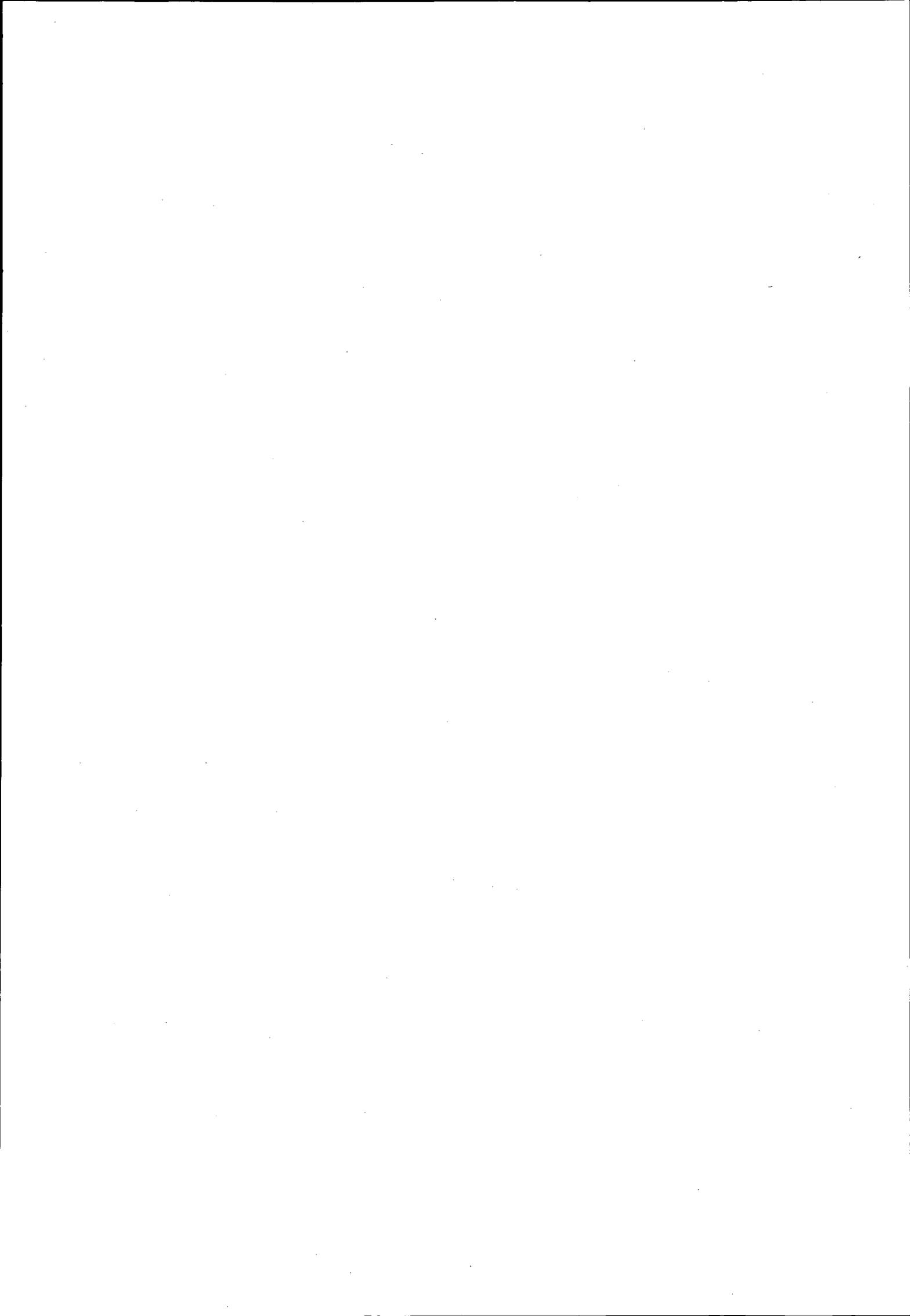
データベース構築促進及び技術開発に関する報告書

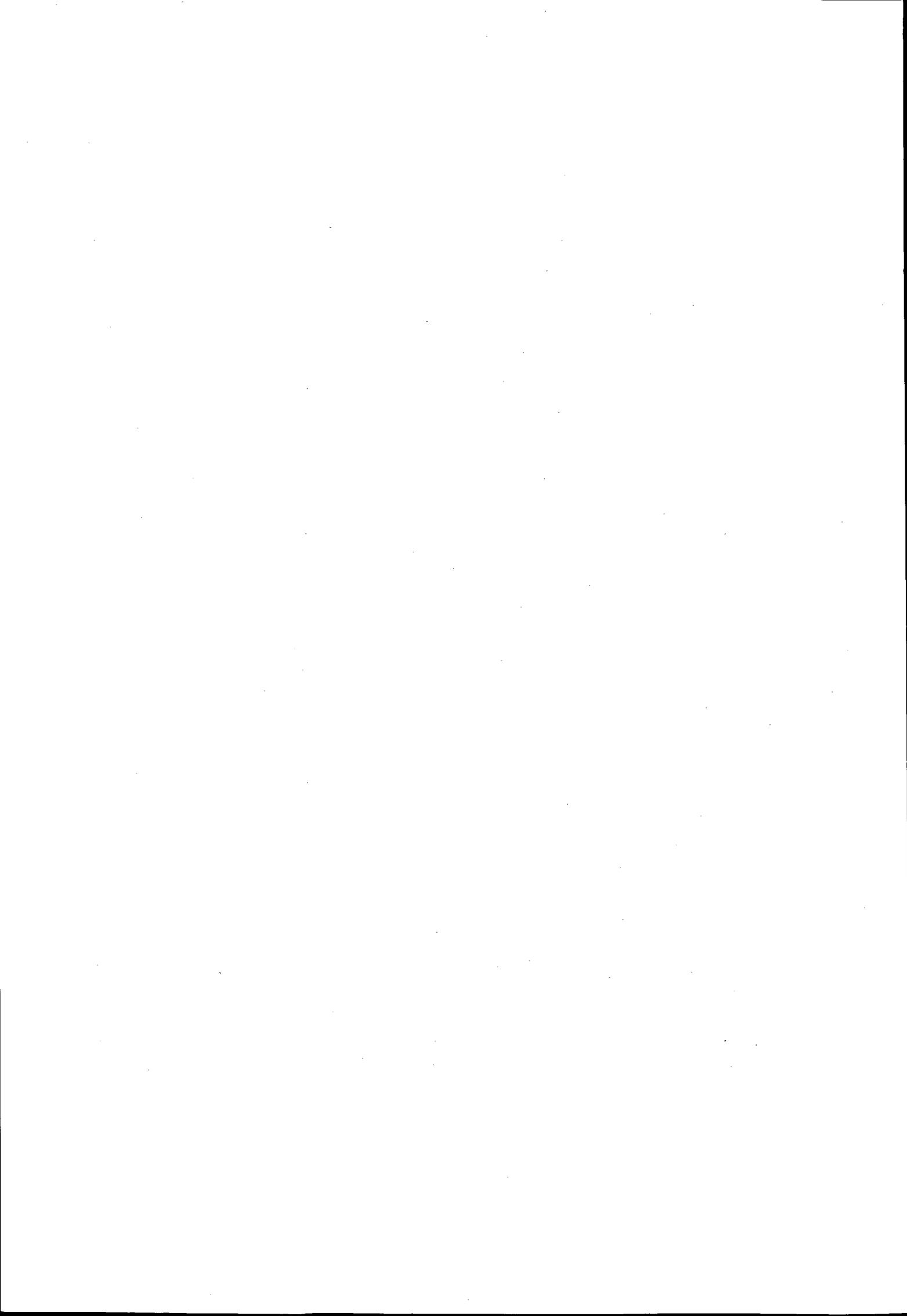
3次元マッピングデータベースの技術開発

平成5年3月

財団法人 データベース振興センター
委託先 (株)日本総合技術研究所

本事業は、日本自転車振興会から競輪収益の一部である機械工業振興資金の補助を受けて作成したものである。



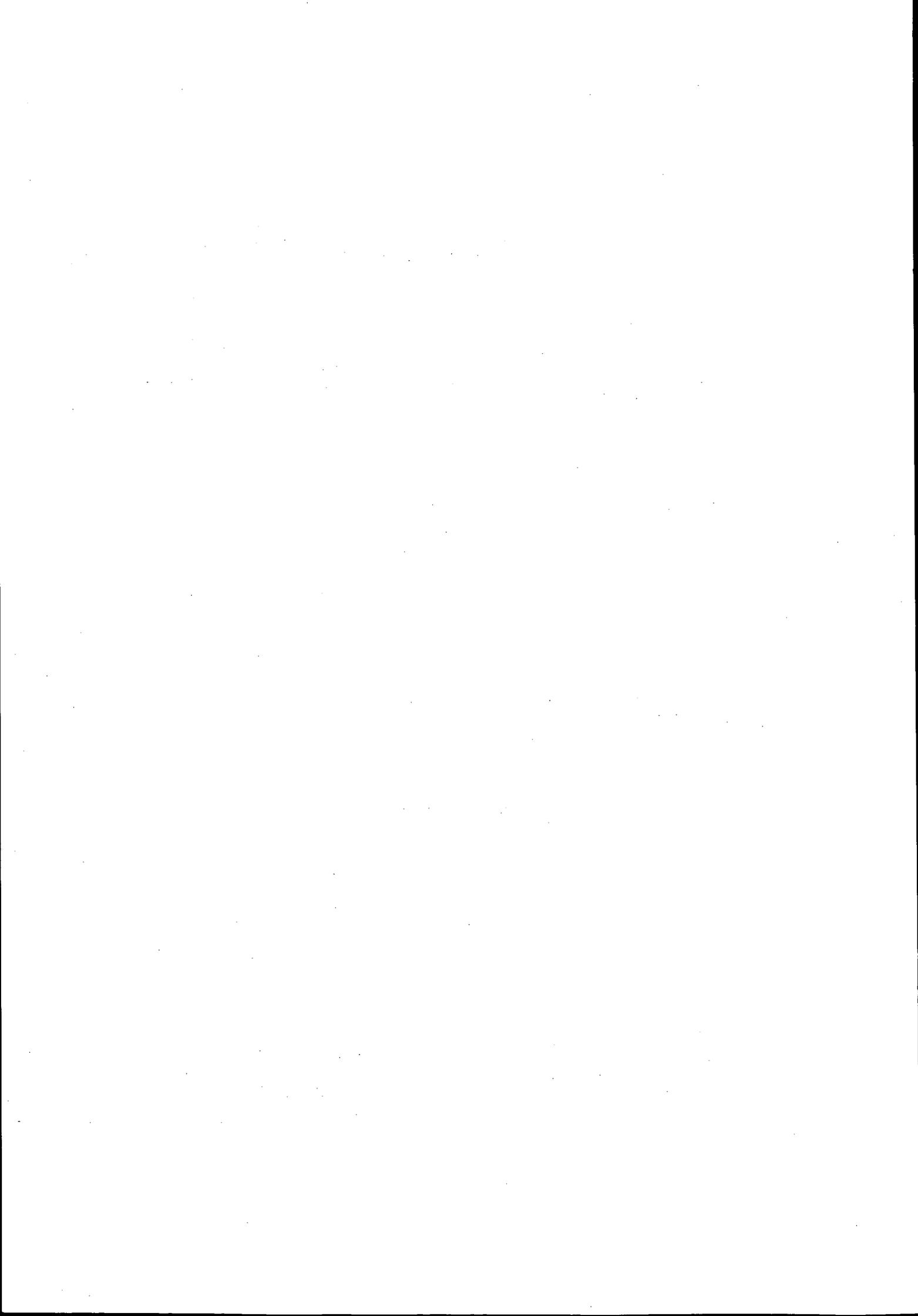


データベース構築促進及び技術開発に関する報告書

3次元マッピングデータベースの技術開発

平成5年3月

財団法人 データベース振興センター
委託先 (株)日本総合技術研究所



序

データベースは、わが国の情報化の進展上、重要な役割を果たすものと期待されている。今後、データベースの普及により、わが国において健全な高度情報化社会の形成が期待される。さらに海外に対して提供可能なデータベースの整備は、国際的な情報化への貢献および自由な情報流通の確保の観点からも必要である。しかしながら、現在わが国で流通しているデータベースの中でわが国独自のものは1/3にすぎないのが現状であり、わが国データベースサービスひいてはバランスある情報産業の健全な発展を図るためには、わが国独自のデータベースの構築およびデータベース関連技術の研究開発を協力を促進し、データベースの拡充を図る必要がある。

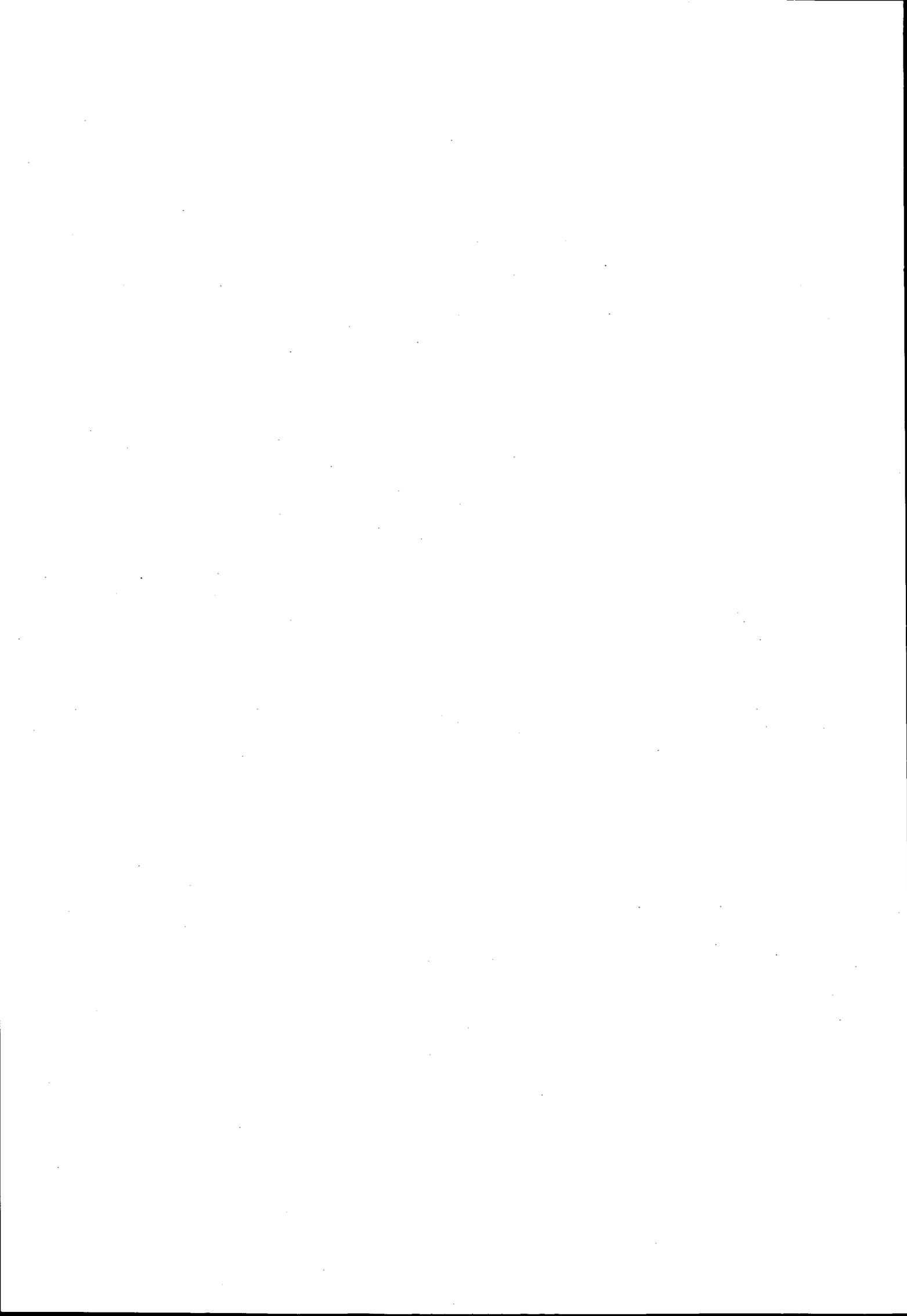
このような要請に応えるため、(財)データベース振興センターでは日本自転車振興会から機械工業振興資金の交付を受けて、データベースの構築および技術開発について民間企業、団体等に対して委託事業を実施している。委託事業の内容は、社会的、経済的、国際的に重要で、また地域および産業の発展に寄与すると考えられているデータベースの構築とデータベース作成の効率化、流通の促進、利用の円滑化・容易化などに関係したソフトウェア技術・ハードウェア技術である。

本事業の推進に当たって、当財団に学識経験者の方々に構成されているデータベース構築・技術開発促進委員会（委員長 山梨学院大学教授 蓼沼良一氏）を設置している。

この「3次元マッピングデータベースの技術開発」は平成4年度のデータベースの構築促進および技術開発促進事業として、当財団が(株)日本総合技術研究所に対して委託実施した課題の一つである。この成果が、データベースに興味をお持ちの方々や諸分野の皆様方のお役に立てば幸いである。

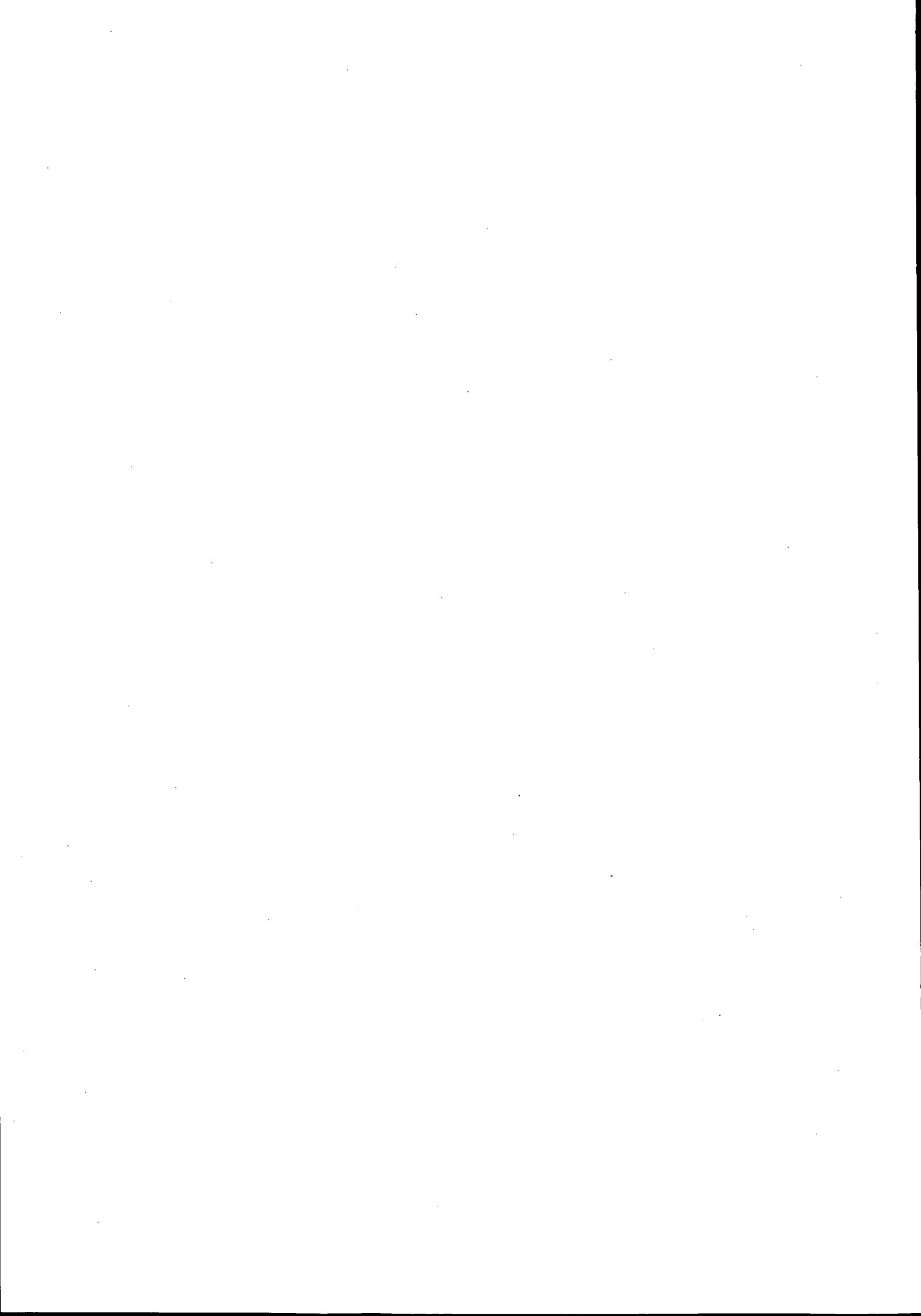
なお、平成4年度データベースの構築促進および技術開発促進事業で実施した課題は次表のとおりである。

平成5年3月



平成4年度 データベース構築・技術開発促進委託課題一覧

| 分野 | 課題名 | 委託先 |
|-----------------|---|---|
| 社 会 | 1 変異タンパク質配列データベースの構築 2 新聞縮刷版見出しデータベースの構築 3 ファジィに関する文献データベースの構築 4 医療用医薬品抗生物質データベースの構築 5 交通事故調査データベースの構築 6 楽器データベースの構築 7 人体計測データベースの構築 8 大学におけるデータベース利用教育システムのプロトタイプ作成 9 先進複合材料データベースの構築 10 博物館所蔵地図資料所在情報データベースの構築調査 | 日本電子計算(株) (株)朝日新聞社 (財)日本情報処理開発協会 (株)小田島 (財)日本自動車研究所 (株)ダイソメディアサービス (社)人間生活工学研究センター 日外アソシエーツ(株) (財)次世代金属・複合材料研究開発協会 (財)地図情報センター |
| 中小企業振興 地域活性化 | 11 地域流通最適化データベースのプロトタイプ作成 12 異分野研究のための知的オリエンテーション・データベースシステムのプロトタイプ作成 13 在宅勤務者サポート・データベースの構築調査 | (財)日本ボランティア・チェーン協会 (株)けいはんな (株)志木サテライトオフィス・ビジネスセンター |
| 海 外 | 14 銅基複合材料日本特許英文データベースの構築 15 技術協力供与機材データベースのプロトタイプ作成 16 先端産業分野における専門用語の電子辞書データベース化の調査研究 17 マーケティングコードの英文データベースの構築 | 神綱リサーチ(株) (財)日本国際協力システム 科学技術情報研究所(株) (株)帝国データバンク |
| 技 術 | 18 安全研究における多重ソース・システム構築のための基本安全用語データベースの開発 19 3次元マッピングデータベースの技術開発 20 データベース検索サポートシステムの調査研究 21 グループウェアにおけるデータベースシステムに関する調査研究 22 パーソナルコンピュータとLANの利用による非定形データベースのプロトタイプ作成 23 知的資源型データベースの調査研究 | (株)紀伊国屋書店 (株)日本総合技術研究所 セントラル開発(株)情報図書館 RUKIT (株)イフ・アドバタイジング (株)メイテック (株)ジャパンコミュニケーションズ インスティテュート |



目 次

| | |
|-------------------------|----|
| 1. システムの目的 | 1 |
| 2. システムコンセプト | 2 |
| 2. 1 利用局面の確認 | 2 |
| 2. 1. 1 都市開発業務 | 2 |
| 2. 1. 2 都市計画業務 | 3 |
| 2. 1. 3 都市管理業務 | 4 |
| 2. 2 自治体における利用の特色 | 5 |
| 2. 2. 1 本システムのユーザ層について | 5 |
| 2. 2. 2 自治体クラスによるシステム利用 | 7 |
| 2. 3 システム特性の設定 | 9 |
| 2. 4 3次元データの所在 | 10 |
| 2. 5 データソースの考え方 | 11 |
| 2. 5. 1 地表面への投影情報 | 11 |
| 2. 5. 2 3次元建造物の表示 | 13 |
| 2. 6 地図データベース情報 | 14 |
| 3. システム主要機能の設計検討 | 17 |
| 3. 1 システム構成 | 17 |
| 3. 1. 1 機能構成の考え方 | 17 |
| 3. 1. 2 基本データベースについて | 18 |
| 3. 2 主要機能の基本設計 | 20 |
| 3. 2. 1 データ変換機能 | 20 |
| 3. 2. 2 各種表示機能の設定 | 22 |
| 4. システム構築の考え方 | 28 |
| 4. 1 プロトタイプシステム構築の手順 | 28 |
| 4. 2 システム構成 | 29 |

| | | |
|--------|-----------------|----|
| 4. 3 | 出力機能の評価 | 31 |
| 5. | 成果のまとめ | 36 |
| 5. 1 | 成果概要 | 36 |
| 5. 2 | 今後の課題 | 36 |
| 補足資料 1 | DXFファイル変換システム仕様 | 38 |
| 補足資料 2 | JISAMライブラリについて | 57 |

1. システムの目的

地図データベースは行政機関、自治体、都市ユーティリティ事業者など各界で利用されており、高度な機能を保有したシステムが多数存在する。これまでの地図データベースは二次元の地図情報のみを管理の対象としてきたが、都市構造の複雑化に伴い従来の面的な扱いでは不十分なものとなってきている。

都市における建造物、道路に限らず都市ユーティリティ（ガス、電気など）もその形状は複雑化の一途をたどっており、3次元形状データベースは地図情報と地表面、建造物などの3次元形状データを統合し3次元マッピングデータベースとすることにより、こうした社会的趨勢に対応する、より実体に接近した地域のデータベースを目指すシステムである。特にニーズの高い都市開発や都市管理用に用途を絞ったデータベースの技術開発を行った。

2. システムコンセプト

都市開発及び都市管理業務においてはその処理すべき地域が広範囲であり、関係者も多数に及ぶ他、扱う情報量も膨大なものがある。このような事情によりいくつかの自治体はコンピュータを用いた地理的な情報管理システムを既に運用している。しかしながら都市空間の高度利用がここ数年の間に著しく進捗するにしたがって従来の面的な地域情報データの管理では開発業務、都市インフラの設備管理業務に支障を来す可能性がある。

3次元マッピングデータベースシステム（以下システムと略す）は立体的な都市空間の可視化を自治体全域にわたって繰り広げることによってこうした点を補い、同時に複雑化した都市インフラをより効率的に管理することを目標とするものである。

このような立場から本システムの利用局面の分析を行い、システム特性やデータについての考え方を整理する。

2. 1 利用局面の確認

3次元マッピングデータベースシステムが必要とされる利用局面は、

- ①都市開発業務
- ②都市計画業務
- ③都市管理業務

に大別されることを確認し、その現状と3次元情報へのニーズを整理した。

2. 1. 1 都市開発業務

都市開発は条件付きの都市計画である。公共施設の補充を行いながら土地利用の転換を進めるため、基盤の用途地域を保存しつつ公共施設を設置し開発用の土地利用制限を定める。このように検討すべき事柄が通常の都市計画に比較して多い。多角的な観点から都市開発を策定し関係者の合意を得てゆくためには一般市民になじみやすくしかもビジュアルな表現を可能にする都市開発業務の支援システムが求められている。3次元マッピングデータベースは多くの点でこの条件を満たしていると考えられる。とりわけ広域の地図データと3次元表示をドッキングすることの意義は非常に高く、

CADやCGなどとは異なる公共性の強いシステムとなりうる。

特に、合意形成について現場の業務を整理する。

公聴会、説明会（地元）など当事者間の調整業務が多数行われるがその資料作製に過大な負担がかかる。これらの調整業務を円滑にこなすためには開発のもたらす影響の客観的測定が必要になる。さらには具体的なイメージが不足気味であるため、双方の意見がかみ合わないことも多々あると考えられる。

①局内調整

各種委員会、議会、関係官庁用の資料を作成する傍ら、関係部局内での意見調整を遂行。そのためのバックアップ資料が必要となるが地図情報ベースのアウトプットで調整が行われる。業務内容としては企画調整が主務となる。

②開発関係者間調整

基本計画、施設計画の原案が策定した時点で、外部業者、コンサル等との連携作業が開始される。その情報交換によって施設計画の細部、都市インフラ計画、環境計画のフレームが確定する。

③対地域住民説明

技術的細部資料は不必要であるが、説得力を持った素材を提示しなければ合意形成が得られない状況となりつつある。開発目標とその内容についての明快な提示、さらに開発プロセスの説明や、開発後のイメージの表示なども重要である。

2. 1. 2 都市計画業務

定常ルーチンで為される都市計画の業務は従来の地図データベース（コンピュータマッピング）で十分であるが、希少化した都市空間をアメニティに配慮しつつ有効に活用するためには、立体的な空間の取扱いによってその用途をきめ細かく設定できることが求められる。これにより良質な都市環境をプランニングすることができると期待される。

①都市計画情報の立体化

都市計画情報を透視法で図示することは定常業務としての都市計画を見通しのよいものに変える可能性がある。このために地表面3次元情報と都市計画情報を組み合わせるという用途がある。

②都市環境情報の可視化

本システムでは具体的に表現できないデータを3次元表示によって可視化する。例えば用途規制地域の変更のもたらす影響のシミュレーションを時系列モデルをもとにコンピュータで模擬実験し、その計算結果、推定容積率、人口、温度やエネルギー需要なども3次元で可視化することができよう。

また、本システムのアプリケーションの一部である景観シミュレーションは本質的にイメージ情報の取扱いが可能であるため、航空写真などイメージ情報の処理ができる。従ってイメージデータの表示、GISデータとの重ね合わせが実現できることも重要な要素である。

2. 1. 3 都市管理業務

ここでいう都市管理業務とは都市活動を支える様々なインフラストラクチャの設備、例えば電力、ガス、鉄道、電信、道路などの設備を集約的に管理するものである。これまでもこれらのインフラを運営する事業者は面的なコンピュータマッピングによって設備を一元管理してきた。管理といっても通常は図面管理が主流であり地図情報そのものはグラフィックユーザインタフェースとしての意味合しかもたないものが多かったし、近年まではそれで十分であった。しかし都市における建造物の巨大化、過密化や地下施設、複合建築のの登場などで面的なデータの管理ではこなせない業務が現れてきた。こうした事態は3次元表示を例えばアイソメ表示などによって段階的に取り入れることで一時的に回避はできる。これは抜本的な解決にならないことは言うまでもない。都市インフラを建造物、地表面を含めて3次元表示することはこうした事態に対して有効である。

こうしたタイプの業務支援とは別に不動産物件の集約管理のような場面においても3次元マッピングは有効であることが予想される。例えばオフィスビルにおける高さ、

向きや眺望などの情報を可視化するのに本システムは最適であろう。

以上のような本システムの利用局面を整理したものが表 2-1 である。

表 2-1 利用局面の種別

| 業務分類 | 利用局面 | 解 説 |
|------|------------|---------------------------|
| 都市開発 | 開発計画支援 | 景観パース作製、事前評価 |
| | 開発の合意形成支援 | CGによるプレゼンテーション |
| 都市計画 | 都市計画情報の立体化 | きめ細かな都市空間管理 |
| | 都市環境情報の可視化 | 日影、アムニティなどの模擬実験 |
| 都市管理 | 都市インフラ設備管理 | 錯綜する都市インフラ設備の ビジュアルな把握 |
| | 都市インフラ図面管理 | 不動産管理などにも適用可能 |

2. 2 自治体における利用の特色

都市開発の条件の複雑さは関係者の多さもさることながら、地域に関連するものの視点に立って評価しなければならないことであろうが最大の問題である。また、開発拠点が地域性の異なる様々な場所に及んでいることにも起因している。

このような都市における開発を支援するコンピュータシステム自体も複雑なものとなってきておりデータベースも例外ではない。本システムではユーザの立場からの使い勝手についても配慮し、ヒューマンインタフェースの設計思想を取り入れることでこれを解決することを試みた。

2. 2. 1 本システムのユーザ層について

自治体では専門のプランナーやコンピュータエンジニアが計画を担当するケースは稀であろう。それに対して都市インフラを管理する都市ユーティリティ事業者などでは専任のグループが用いることになる。このように本システムには幅広いユーザがありえる。ここではそれを踏まえてシステムの性格、特にマンマシンインタフェースを検討する。

①ユーザレベルの想定

本システムは専門家向けの建築CADやCGとスタンスを異にし、専門知識は最低限ですむように設計されるべきであるとする。そのためには次の三つのコンセプトを包含する設計ガイドラインを定めることが肝要であるとする。

- ・セルフコンテインド
- ・GUI
- ・エルゴノミクス

これらのコンセプトを概説する。セルフコンテインドとは作業に必要な情報はすべてシステムが保有していることである。そのためには必要最低限の関連情報、例えば都市インフラ設備の属性情報や都市関連情報などを常時参照できるようにすることが求められる。また、入力データをミニマムなものにとどめ、本システムのデータ更新などにかかる人手を減らす配慮ことも必要であろう。オンラインヘルプやチュートリアルの実装も考慮しなければならない。

このような前提のもとにオペレーターから、専門のプランナーまで対象ユーザ層を拡大できる。GUI、エルゴノミクスについては項を改めて述べる。

②エルゴノミクスの必要性

エルゴノミクス（人間工学）の原則をコンピュータ上に取り込んだグラフィックユーザインタフェース（GUI）は総合ビジュアルシステムを標榜する本システムに不可欠な要素である。特に、マルチウィンドウ（リサイズ可能、オーバーラップタイプ）、各種のアイコンやボタンなどはローエンドのコンピュータでも常識となりつつある。都市開発や都市管理の業務に即したアイコンやボタンのデザインも有効であろう。

この他、エルゴノミクスではシステムの間接的要素（ヒューマンファクター）として表2-2の諸要素にたいする吟味を要請している。

表2-2 ヒューマンファクターの諸要素

- ・メニュー選択
- ・直接操作
- ・応答時間と表示速度
- ・対ユーザへの表現方法（メッセージ、スクリーン設計、色、ウィンドウなど）
- ・オンラインヘルプ&チュートリアル

2. 2. 2 自治体クラスによるシステム利用

本システムは自治体関連ユーザを想定しているが自治体のクラス、規模によってシステムニーズや利用方法について相違が存在することが予想される。一般的には以下に記述されるような状況である。

①政令指定都市

政令指定都市において地域管理を掌握する部門が、開発計画の策定に用いる場合を想定。特に都市開発の進行に際しては非常に多くの専門家・市民・有識者との検討・合議がなされるのが通例であり、さらに関係事業者や関係他部署、中央官庁との調整作業も大量の情報処理が強いられる。また、合意形成の過程においては、ビジュアルプレゼンテーションが常識となりつつある。これらの業務を全面的に支援する情報処理の仕組みが早急に必要となる。

②中小自治体

政令指定都市に比較して都市開発は頻度こそ少ないが計画策定作業の複雑さは変わらない。しかしながら、都市景観を評価する機会は希であり3Dデータを常時メンテナンスすることは必要ないと考えられる。

以上のように実際のシステム機能について、ユーザによって要求する機能の精度が異なることが分かる。

政令指定都市とそれ以外の市町村ではデータコンバート機能に対するニーズにかなりの相違がある。前者では関連システムが多数存在し、その情報蓄積もかなりのものがある。従ってデータの流用も頻繁に起きるのに対し、後者では高々メジャーなGISとCADへの対応さえ装備されていれば充分であると予想される。他方、イメージ画像処理機能に対しては完全にケースバイケースである。

3Dマップの内容については次のように考える。

一般の中小自治体では都市形状の3Dデータを常時メンテナンスすることは負荷が大きすぎる。しかしながら標高データをGISと連動できる形式で保持して置く事は有用である。標高データそのものは特別なケース（土地の異常隆起、広域の陥没、地滑りなど）でない限りメンテナンスの手間は少ない。

また、その用途は都市景観に限らず環境保全、防災、各種デモンストレーションに活用できるからである。政令指定都市では都市景観の精度を高めるためにも両者のデータが重要であることは言うまでもない。これにともない3D表示機能もややレベルの差を持たせたものになる。3Dのモデリング機能、レンダリング機能、アニメート（ウォークスルー）機能はいずれも政令指定都市では専用ソフトで高性能のものが要求されるのに対し、それ以外の中小自治体では簡易的なもので充分であると見られる。

2. 3 システム特性の設定

想定ユーザの利用場面の検討を踏まえて3次元マッピングデータベースがどのような特性を具備すべきかについて整理する。

既製システムとの機能分化

コンピュータマッピングは都市計画、都市管理などに深く浸透し、それぞれの業務に応じた機能、性能、形態をとっている。新システムはこれと役割分担を行う必要がある。また、コンピュータマッピングの機能補完も考慮すべきであろう。

ハードウェア、ソフトウェア（OS、データ等）の資源活用

新システムではハード、周辺機器の共有をはじめた設備の共有を重視する。また既製システムのデータ蓄積も無視できない。このハード、データ資源を有効に利用できるシステムが望ましい。

運用方式&メンテナンスの容易性

3次元マッピングデータベースシステムの表示対象となるデータはきわめて多種多様であり、データ更新作業が膨大になる。本システムが恒久的に利用されるためにも可能な限り自動的かつ最小限にデータ更新できる仕組みが求められる。

ユーザへの配慮 — 誰がどのように使うか —

本システムのプラットフォームとしてEWSの導入可能性が高いがシステムの操作そのものはUNIXの最低限の知識で活用できるものでなくてはならない。UNIXの持つ機能、操作はかなり複雑なものである。GUI環境はヒューマンインタフェースデザインへの配慮がまだ不十分な面があると考える。

2. 4 3次元データの所在

国土地理院の国土数値情報が最も利用しやすい。建設省国土地理院では標高に関する情報として等高線データと格子型のメッシュデータを用意している。3次元マッピングはこれらの情報を取り込んで従来の地図データベースの情報と一体化することが可能である。

①等高線データ

等高線のデジタルデータは縮尺1:25,000の地形図用のものを容易しており、1:25,000の地形図の作図用の用途が多い。等高線データは線ベクターの集合という形態をとっている。ポリゴン情報としての扱いはなく点情報間をスプライン曲線で補間するといった手法は用いていない。

②格子型メッシュ標高データ

格子型メッシュの標高データは、縮尺1:25,000の地形図をスキャンして、200×200のメッシュを切り等高線をもとに計算して求められている。これらの作業は国土地理院の開発したアプリケーションで自動化されている。

メッシュ間隔は50mおきとなるが、この間隔は地形が複雑な場合には問題があるが、読み込む地形図の精度、ゆらぎ等から判断すると妥当であると考えられる。ただしメッシュ間隔はアプリケーションの機能で任意設定が可能となっている。

標高データは等高線のベクターデータよりも格子型のメッシュデータの方が応用に適しており需要も大きい。そのため標高の格子型メッシュデータを現在公開している。

2. 5 データソースの考え方

ここでは既存のコンピュータマッピング保有データのうちで本システムに取り込むべき情報を選定する。システムの特性にリストアップしたように本システムはコンピュータマッピングの地図データベースとのデータ交換に重点を置く、その基本はデータソースとしての取り込みである。きめ細かな都市計画もしくは都市管理情報を設定するためにはこれらの情報の取り込みは不可欠である。データ変換さえ構築できれば自動的に3次元表示がある程度まで保証されることでもある。ここで都市計画関連情報を例にとればそうした情報として次の三種類がある。

2. 5. 1 地表面への投影情報

規制界種データとは土地の用途規制の境界線を示したものである。このデータは原則的に曲線情報である。この曲線データを地表面に投影する手続きはサーフェスマッピングと呼ばれる手法が適しているが、表現精度の限界もある。

●規制界種リスト

| | |
|---------------------|-----------------|
| 用途地域界（建坪率、容積率、高度地区） | 緑地界 |
| 日影規制既製区域 | 墓園界 |
| 外壁後退距離 | 火葬場 |
| 特別工業地区界 | 下水処理場 |
| 高度利用地区界 | 汚物処理場 |
| 特定街区界 | 塵焼却場 |
| 防火準防火地域 | 学校 |
| 風致地区界 | 駐車場 |
| 駐車場地区界 | 団地住宅施設 |
| 臨港地区界 | 土地区画整理区域界 |
| 緑地保全地区界 | 市街地再開発事業区域界 |
| 促進区域界 | 地区計画界 |
| 都市計画道路界 | 再開発地区計画 |
| 公園区域界 | 首都圏整備法における既成市街界 |

土地利用現況図区分は地表面への写像（マッピング）を行うタイプのものである。このデータのうち住宅、施設データは建物現況図のうちで取り扱うため景観シミュレーション中では表示する必要はないものと考えられる。従って下記の項目中で☆印を付けたものについてのみ地表面への写像を行う。他の区分情報については適宜サーフェスマッピングを行う。

●土地利用現況図区分

| | |
|--------|-------|
| 専用住宅 | 公官庁施設 |
| 共同住宅 | 都市施設 |
| 商業併用住宅 | 公園施設 |
| 工業併用住宅 | 緑地施設 |
| 業務専用施設 | 墓地施設 |
| 店舗専用施設 | 未利用地 |
| 宿泊施設 | 駐車場施設 |
| 娯楽施設 | ☆鉄道 |
| 遊技施設 | ☆道路 |
| 重工業施設 | ☆水田 |
| 軽工業施設 | ☆畑 |
| 倉庫施設 | ☆山林 |
| 運輸施設 | ☆湖沼 |
| 教育施設 | ☆河川 |
| 医療施設 | ☆運河 |
| 厚生施設 | ☆堤防 |
| 宗教施設 | ☆原野 |

2. 5. 2 3次元建造物の表示情報

建造物のディテールを表現する場合は建物種別毎にテクスチャを割り当てる。このテクスチャは予め妥当なデータをライブラリに登録して置き3次元表示させる際にこの属性をシステムが自動的に結合する。

●建物現況図

| | |
|--------|----------|
| 専用住宅 | 公官庁施設 |
| 共同住宅 | 都市施設 |
| 商業併用住宅 | 公園施設 |
| 工業併用住宅 | 防衛施設 |
| 業務専用施設 | 農業施設 |
| 店舗専用施設 | サービス工業施設 |
| 宿泊施設 | その他 |
| 娯楽施設 | |
| 遊技施設 | |
| 重工業施設 | |
| 軽工業施設 | |
| 倉庫施設 | |
| 運輸施設 | |
| 教育施設 | |
| 医療施設 | |
| 厚生施設 | |
| 宗教施設 | |

2. 6 地図データベース情報

コンピュータマッピングの保持する地図データベースの内容を明確にすることによってその管理データの3次元マッピングデータベースとの関連を整理する。地図データベースは地理空間および非地理空間要素の関係はテーマと呼ばれる情報として総括され管理されている。。それらには特性テーマ、ネットワークテーマ、面テーマ、ラスタテーマの4種類のテーマがある。これらの内容については以下にまとめるが本システムではその取り込み範囲を面テーマとネットワークテーマに限定した。

(1) 特性テーマ

特性テーマと呼ばれるデータはひとつ以上のデータベーステーブル内における属性セットに個々に関係する点、線、弧、あるいはテキストを含んでいる。

(2) ネットワークテーマ

このデータはネットワークと呼ばれるトポロジカル構造における終点と終点の結合である線形特性により構成される。ネットワークトポロジーではリンクとノードを含んでいる。リンクは互いのネットワークを結合する属性である。ノードはリンクの結合点であり、線形特性間の連結性を定義するものである。例えば、リンクは道路、地下埋設パイプ、ケーブルなどを表現し、ノードはこれらが互いに結合される点（道路の交差点など）を表現する。都市管理情報としてこれらは重要である。

(3) 面テーマ

面テーマデータは閉じた領域をカバーするベクターベースのポリゴンにより構成される。ポリゴンは、自治体内における個々の街区情報、不動産区画といった属性に関連した均一な地域を表現している。面テーマ内のポリゴンは重ならないのが常で、各閉領域は一回のみカウントされる。

(4) ラスタテーマ

ラスタテーマは、あるカバー範囲を画素と呼ばれる小さな格子セルのマトリクスに分割することにより生成される。各画素として地表面上のある位置における面に対

応する。非常に数多くの等サイズ画素が利用されており、一般にはラスターテーマでは百万単位の画素が含まれている。

ラスターテーマはひとつ以上のオーバーレイから構成されラスターテーマ内の各オーバーレイは同じ地理的空間および同じ空間的解像度を有していなければならない。各オーバーレイ内の画素は特定の現象に関連した属性値を含んでいる。

さらに地図データベースはテーマデータのファイル管理を次のように行っている。これらのファイルはデータベース用図面ファイルを蓄積するために存在しシステムにより管理される。

(1) テーマ定義ファイル

データベース用に定義されたテーマに関する情報はテーマ定義ファイルに蓄積される。

(2) トポロジファイル

面テーマに対する構造ブロックを形作るネットワークやポリゴントポロジに関する構造的情報はファイル登録される。このファイルもまた、各特性、ネットワーク、面テーマとともに特性情報を記録する識別子を含んでいる。

(3) ノードファイル

ネットワークテーマあるいは面テーマに対するトポロジを生成する際、システムは各ノードに付属する結合に関する情報を記録するためのノードファイルを自動的に生成する。

(4) ポリゴンインデックスファイル

テーマ内のデータ検索やデータ抽出など、データ更新のスピードを上げるために、システム内で利用される。

面テーマに対するポリゴンインデックスを任意に生成することができる。このインデックスはテーマ内における各ポリゴンの空間的広がりを示すリストを管理するものである。

(5) ラスターオーバーレイファイル

ラスターテーマ内の各オーバーレイは、それぞれ分離されたファイルとして蓄積される。

3. システム主要機能の設計検討

3. 1 システム構成

システム特性を機能構成の観点から整理し、技術動向に対応したシステムを目指すための特色を鮮明にした。また主要なデータベースの内容について検討し、機能構成を具体的に検討した。

3. 1. 1 機能構成の考え方

本システムは最新のハード、ソフトのコンピュータテクノロジーを最大限に取り入れたものであるべきである。とりわけ次のような特色を持たせることによって差別化を図ることが望ましい。

- ・汎用性 3次元表示に重点を置くことはいうまでもないがその適用範囲が都市情報の広汎な局面をカバーできるものでなければならない。特に都市インフラの要素を表現できることが鍵となる。
- ・将来性 ハードウェアの進歩は日進月歩であり、アーキテクチャが旧弊なものとなりがちである。ハードウェアは時代の主流となるものであることが好ましい。
- ・協調性 既成システム、特にコンピュータマッピングとの情報統合を図ることが重要である。
- ・容易性 可能な限りマニュアル不要、メンテナンスフリーであること。操作の容易性も当然考慮されるべきである。

3. 1. 2 基本データベースについて

都市景観の基本的な3次元情報として必要なデータについてリストアップすると共に内容についても整理する。

(1) DTM (Digital Terrain Model) データベース

地形情報の三次元モデルをデータベースとして管理するシステム。管理対象としては自治体全域を持つことが望ましい。地形(標高)情報は原則的に暦年変化がわずかであり更新の手間はそれほどかからないため3次元データの蓄積源として保持するのは合理的であると考えられる。ただし造成等の細部修正の必要がある。その用途について整理する。

① DTMデータの必要性

- ・デジタル地形図は国のDBとして共通財産である。このDBのための標高データを自治体などの公共もしくは公益事業者で管理することが求められるのは当然である。
- ・都市ユーティリティ事業では地表面をベースに地下埋設したり、地上敷設を行うことが多いためニーズが高い。
- ・ダムのサイトプランニング、道路設計などの土木設計では路線決定、切り土、盛り土を算定することが必要となる。
- ・景観デザイン、景観のシミュレーションのための地形表現を提供する。
- ・標高を他の連続量と結合することでトリップタイム、コスト、地価、人口、地価水位、商圈などを詳細に表現できる。

② DTMデータ表示

DTMデータの表現は3次元のマップを作製する時に顕著な影響を与える。その詳細は「4. 3」で論じられるが、その表示方法としては以下にあげるような分類がある。

- ・メッシュモデル

矩形状の格子で空間情報を区切るタイプのもの。通常の地図作製ではこれがベースになることが多い。

- ・T I N (Triangulated Irregular Network)

三角形不規則ネットワークによる表示方式である。この特徴としてはメッシュモデルに比較して冗長度がないが傾斜度などの計算が容易である。

- ・コンターモデル

いわゆる等高線モデルである。三次元化の場合、階段上になったステップモデルとこれを三角メッシュ化したものが三角メッシュコンターモデルがある。

(2) 境界情報データベース

行政区画として市町村区界、都道府県境などの境界線情報を地表面に投影したものの。従って、3次元曲線データとなる。この具体的な内容については「2.5 データソースの考え方」において検討された。

この境界情報は2次元においても完全にポリゴンとして整備、公開されているものはわずかである。こうしたことから3次元化には多大な困難、例えば多大なロイヤリティや精度のズレなどの障害がある。広域にわたってマップデータを整備するためには公的な機関に依存することになる。

もっと特殊用途でニーズのある規制界種情報については「2.5.1」で整理した。

(3) 都市計画2D情報データベース

都市計画情報システムのデータを三次元へ投影するためのものである。もしくは三次元オブジェクトとするためのベースデータとして用いることもできる。内容としては

- ・道路、河川、堤防、運河など
- ・鉄道
- ・建造物（地上）

・地下埋設物など

これらのあるものは通常の地図データベースと重複しているが都市計画業務ではこちらのデータを優先させることはいうまでもない。というのは一般の地図データベースの保持するベースマップとはデータ源、精度が異なる場合があるからである。

3. 2 主要機能の基本設計

3. 2. 1 データ変換機能

二次元系のデータは自治体が蓄積している傾向が強い。ただしそれらはデジタル化されているとは限らない。他方、三次元系のデータは外部業者（都市ユーティリティ事業者、デベロッパー、コンサル）などが開発プロジェクトに対応して一時的に生成するものであるが、3次元マッピングデータベースシステムは本機能によって両者を結合し、これまで外部業者などに依存していた機能の一部自営化も図る。

三次元形状データ（3D都市モデル）を扱う環境を本システムは標準的に保持することとする。これにより、従来のシステム群をサポートできよう。また、外部データの取り込み（Import）、吐き出し（Export）が本システムの機能として具備されることが不可欠である。

これらの変換機能のうちプロトタイプシステムでは3D交換フォーマットとしてのDXFファイルコンバート機能を新規作成する。これによって一部制限があるものの地図データベースの2D地図情報に高さ情報を付加するだけで3次元化できる。さらに高さ情報を

付加するためにUNIX上で開発したISAMライブラリを適用した。

ただし、地図データベースの保有するラスタ情報や属性情報については実用化システムの段階で組み込むこととした。

3次元マッピングデータベースシステムはオリジナルのファイルの他に次のようなGIS系のARC/Info、CAD系のDXFデータなどの読み込み、書き出し機能が要請される。ただし、プロトタイプではARC/Infoでなく、弊社の保持する地図データベース情報（aim）を用いた。

ARC/Info

GIS*の事実上の標準交換ファイルである。従ってこれをサポートしているGISパッケージは非常に多い。ここでのレコード構造はトポロジカルデータ構造と呼ばれており、GISに特化した特殊なファイル形態である。

*GIS (Geographic Information System) : 地理情報システム。地図データベースとほぼ同義である。

3D交換フォーマット

DXF、IGESのどちらかをサポート。現状ではDXFがメジャーである。DXFはCADソフトの一種であるAUTOCAD (AUTODESK社製) のファイル形式である。テキストファイルであるためUNIX、MS-DOS、Macといった異なったOS間でのデータ移行が簡単にできる利点がある。ただし、バージョン毎にかなり異なった仕様であり混乱がある。最新版は10.3である。

SQL

属性情報を管理するための汎用データベース言語仕様である。IBM、DECなどのメインフレームはこれをサポートしており属性情報を管理する場合には不可欠なものである。

ラスターベクターデータ変換

CG系のデータはラスターデータでありこのままではGISなどのベクター系のデータとの共存は難しい。現在両者のデータ形式の変換技術は実用に耐え得るものが確立しており、ラスターベクターデータ変換機能を取り込むことは必須のものであるといえる。さらに変換前のラスターベクターデータを重ね合わせて表示できる機能も必要である。

3. 2. 2 各種表示機能の設定

本システムの表示部分の基幹機能のアウトラインをまとめる。ここでの記述は3次元CGソフトの一般的な機能とかなり重複する部分もあるが、開発スタンスとしては広域地域情報を対象にしたシステムであることを念頭において進められた。実用システムではより詳細な検討によって機能の記述を掘り下げる必要がある。プロトタイプシステムでは重要度の高い機能だけを簡素化して取り上げた。

(1) 2D-3D描画機能

モデリング機能

主として基本モデルの修正と開発該当区域のの全体フレームの中に取り込むなど付加的に用いる場合が多い。このため機能深度はそれほど求められない。モデルタイプの基本はソリッドモデルを採用する。ソリッドモデルは加工柔軟性におり、ブーリアン演算のような図形の組み合わせ処理が有効であるためCAD系でも主流になりつつある。

基本的なメニュー構成としては次のようなものを設定した。

- ①ファイリング機能（新規、保存、読み込み、インポート、エクスポート（インポート、エクスポートについては「3. 2. 1」参照））
- ②2D描画機能（直線、円、楕円、多角形、Bスプライン）
- ③3D描画機能（平面、円柱、楕円柱、多角柱、回転体、Bスプライン曲面など）
プロトタイプではこれらの機能はベースマップとCADの方に完全に依存することとする。実用システムではこれらを一体的に保持することが望ましい。
- ④編集機能（移動、選択、複写、消去、拡大、縮小、修正、回転）
プロトタイプではこれらの機能はベースマップとCADの方に完全に依存することとする。実用システムではこれらを一体的に保持することが望ましい。

⑤オーバーレイ（新規、アクティブ、ノンアクティブ、消去）

地図データベースのデータを変換する場合全ての画素を表示すると編集や物体の判別が困難になることがある、これを避けるためにオーバーレイ機能（CADではレイヤーとも称する）を持たせる。プロトタイプではこれらの機能はベースマップとCADの方に完全に依存することとする。実用システムではこれらを一体的に保持することが望ましい。

⑥簡易表示（ワイヤフレーム、陰線処理）、表示平面切り替え

モデルを修正する場合にそのデータを表示しておかねばならないがその画質が高度のものであると画面表示に多大な時間がかかる。これをさけるために線画レベルの表示を行う。

⑦印刷機能（ページプリンター、プロッター出力）

ページプリンター出力のみ可能。プロッターは将来的にオプションとして用意する。

⑧数値入力、ディジタイザー入力、マウス入力

今回はサポートしていないが実用システムでは不可欠な機能となる。これによってデータ入力を行うケースが非常に多いからである。

(2) レンダリング機能

いわゆる景観シミュレーション作成機能がこれに対応する。視野、視点と移動軌跡を自由に指定し景観を動的または静的に表示することも含まれる。

①レンダリングレベル

レンダリングはアウトプット用の画像生成を行う機能である。画質にはいくつかのレベルがあり、そのレベルに対応した手法の代表例を挙げると

- ・ワイヤフレーム
- ・Zバッファ
- ・フォング

- ・レイトレーシング
- ・ラディオシティ

のようなものが典型である。これらは上から順に画質が高くなる。このうち複数をサポートすることを目標とする。ここではワイヤーフレーム表示とZバッファをテストおよびワーク用、レイトレーシングおよびラディオシティを完成画面用のレンダリング手法として選定する。ただしこの両者はハードウェアの性能、特にCPU性能によっては実用にならないほど遅くなることもあり、本格稼働システムでのデータ構造設計、アルゴリズム設計には多大な努力を必要とするものである。

②レンダリングサイズ

CRTのサイズに合わせて任意サイズ指定が可能であることが望ましい。ただし、CPUの性能によっては19インチサイズの画面を作成するために過負荷である場合があり当面標準的なものとなりつつある640×480で運用する。

③マッピング機能

ここでいうマッピング機能とは

テクスチャーマッピング

ピクチャーマッピング

バンプマッピング

イメージマッピング (サーフェスマッピング)

など3次元物体(地表面、道路、建造物など)の光学的属性や表面の属性を指定できるのがCGソフトでは通例であるが3Dマッピングにおいては高度な機能は必要としない。ただ前述の地理的な境界情報や土地利用情報は本機能の一部によって地表面に写像する。地理情報の張り付けはサーフェスマッピングによって行う、これはイメージマッピングの拡張であり地図の位置情報の補正を施したデータを地表面に張り付けるものである。

⑤テクスチャライブラリ

建造物など3次元マッピングのオブジェクトに対する属性データライブラリを予め登録管理できるシステムが

(3) アニメーション機能

連続的な画像を予め設定したシナリオにそって自動生成する。動線に沿った視点移動はこの機能の一部であるとも考えることができる。

キーフレーム方式

フライスルー機能

カメラオブジェクト機能

ライトオブジェクト機能

インビットウィーン機能

プレビュー機能

テクスチャ張り付け機能

コマどり装置対応

これらの機能は内部モジュールとして実用システムに組み込むべきものであるが、プロトタイプシステムではMMD (MacroMindParacomp社製MacroMindDirecter) などの動画作製ツールに依存し、外部モジュールとして評価した。

(4) シミュレーション機能

シミュレーション機能は形態制限表示機能、連続視点移動機能に大別される。プロトタイプシステムでは本機能は組み込まれないが地図データを3次元化し、それを標準フォーマット (DXFなど) に変換して外部モジュールとして評価することは可能である。ただそれらはニーズに真に対応していなものが多いため、ここでは実用システムでの導入スタンスとしての機能概要をまとめた。

◎形態制限表示機能

形態制限の綿密な検討は開発者に委ねるのが原則であるがラフな評価が本システムで対応可能である。

①容積率評価

容積率とは建築物延べ面積にたいする敷地面積の割合のことである。これを個別の建築物について簡単に計算できることが望まれる。特定街区あたりの平均容積率も計算する場合がある。

②建坪率評価

建坪率制限は形態制限の一つであるが、これも用途地域ごとに原則の数値基準がある。その計算アルゴリズムは容易であるため例外の処理なども組み込まれていることが望ましい。

③斜線制限表示

立体的なボリュームの制限を基底するのが斜線制限である。道路斜線（前面道路による）、隣地斜線、北側斜線などとともに適用規則がカスタマイズされて、計算結果も3次元的に表示されることが望ましい。

④日影規制表示

自治体の地理的座標と日時の指定によって特定建築物の日影規制ラインを計算し表示する機能。

◎連続視点移動機能

ウォークスルーは建築CAD、CG系では常識的な機能となりつつある。このため導入のポイントは機能の有無ではなく、視点移動機能の画像の品質とレスポンス、使いやすさとなろう。なおこの機能はアニメーション機能とかなりオーバーラップする部分があるがこちらの機能はより一層都市管理、都市計画支援などに特化したものを想定する。

(5) プレゼンテーション機能

プレゼンテーション機能として一般的なものを場面による分類をもとにして整理する。プレゼンテーション機能はある意味で出力機能の一部である。印刷機能と画面表示機能がそれである。前者はハンドアウトなど印刷物やその延長としてのOHPによるものであり、後者はそれほど一般的ではないがディスプレイ（CRT）によるものがある。プロトタイプではCRTとハードコピーでの出力をサポートするものとした。

◎印刷（ハンドアウト）によるプレゼンテーション

印刷は必ずしも自前のプリンターによるものでなくとも良い。カラー出力の場合は出力サービスを利用することでも対応可能である。印刷機能は高度なものは必ずしも高度な機能を必要としない。モノクロ印刷の場合、グレースケールによる精細な印刷ができることが望ましい。A3までのサイズが最低限必要となる。他方カラー印刷の場合、フルカラーの印刷機は高価であり利用頻度も比較的少ない。しかしパンフなどの配布での利用は考慮すべきである。幸いDTPが行き渡りデジタル画像の外注印刷はかなり楽にできるようになった。従ってフルカラープリンターは必ずしも導入する必要がないがDTPへの出力機能は前もって取り込んで置くことが望ましい。

◎ディスプレイによるプレゼンテーション

ディスプレイ（CRT）によるプレゼンテーションは機会としてそれほど多くないと考えられる。外部関係者に対しては主としてデモ画面としてCRTを利用することが多く、当面は社内調整や局内調整が利用場面となろう。しかしながら長期的に見てISDNなど大量のデジタル情報を伝送できるネットワーク、公共のプレゼンテーションセンターなどの普及などが予想できるためCRTによるリアルタイムなプレゼンテーションもシステムの拡張性の一環として見込んで置くことは無意味ではないと見られる。

4. システム構築の考え方

4.1 プロトタイプシステム構築の手順

プロトタイプシステムは機能構成を簡素なものとするためにモジュール化し、各モジュールのデータをDBMSでハンドリングするために最終的に一括する。また、プロトタイプ構築に際しての3次元情報の作成は以下のような簡易的な手続きによった。DXFファイル変換部分を新規に作製し、ISAMによって高さ情報を建物などに付加する。これによって実用化システムでも同じ手続きを踏襲することができよう。

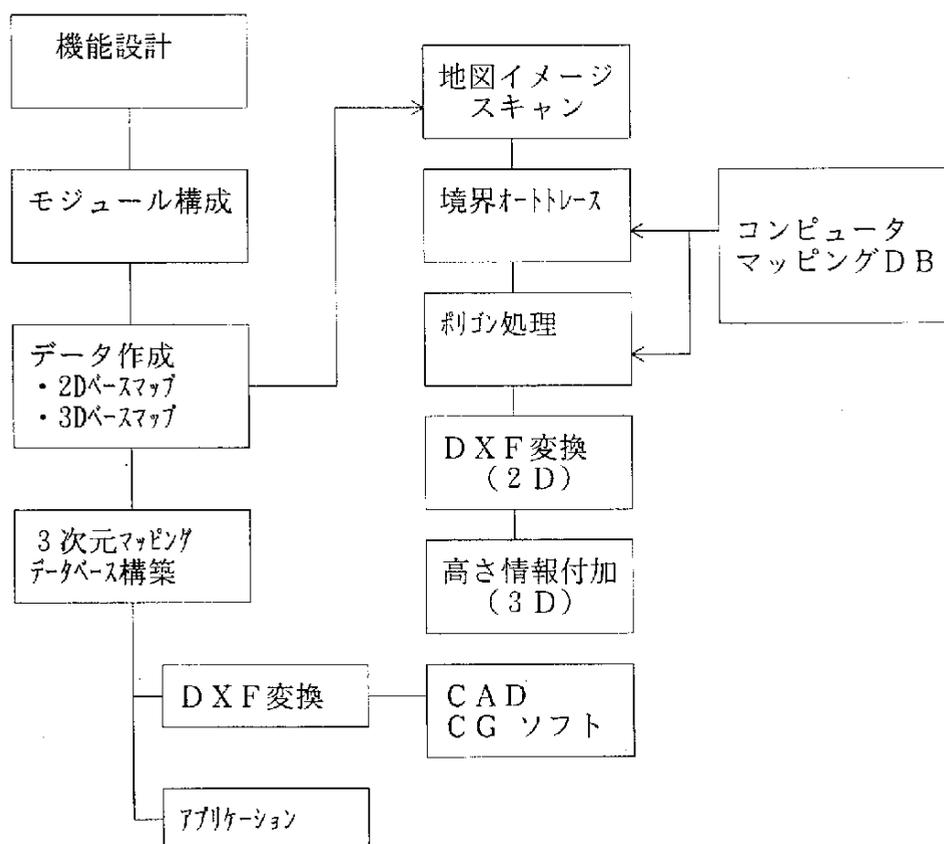


図4-1 システム構築手順

4. 2 システム構成

ハードウェア構成

地図データのサーバーとしてVaxstationを設置しここで3D用のデータ変換を行う。LANを通じてそのデータはMacintosh (Mac) に送られプロトタイプシステムで加工される。

システム本体はMac II fxに置かれ、適宜並列処理を他のCPUに担わせる。

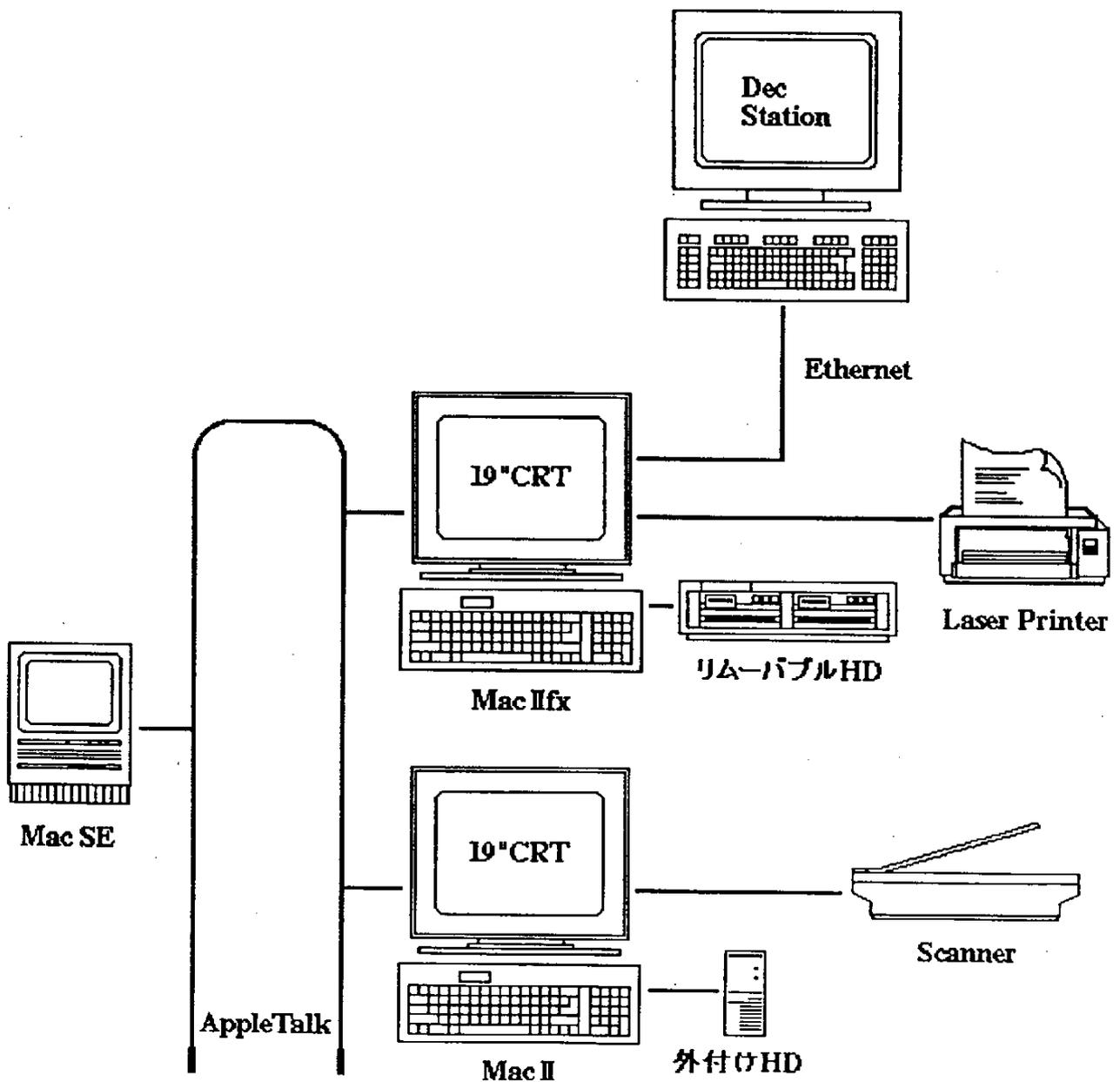


図4-2 ハードウェア構成

ソフトウェア構成

データベースの構造的な関連は次のようなものである。3次元マッピングデータベースの本体は標高モデル、境界情報、地上建物、埋設物などのデータを管理する。DTMは標高データから3次元地表モデルを生成し、同じように3D表示機能はDXFデータを取り込んで景観表示などを行う。このプロトタイプでは地図データ、標高データを全てDXFに変換する構成になっている。

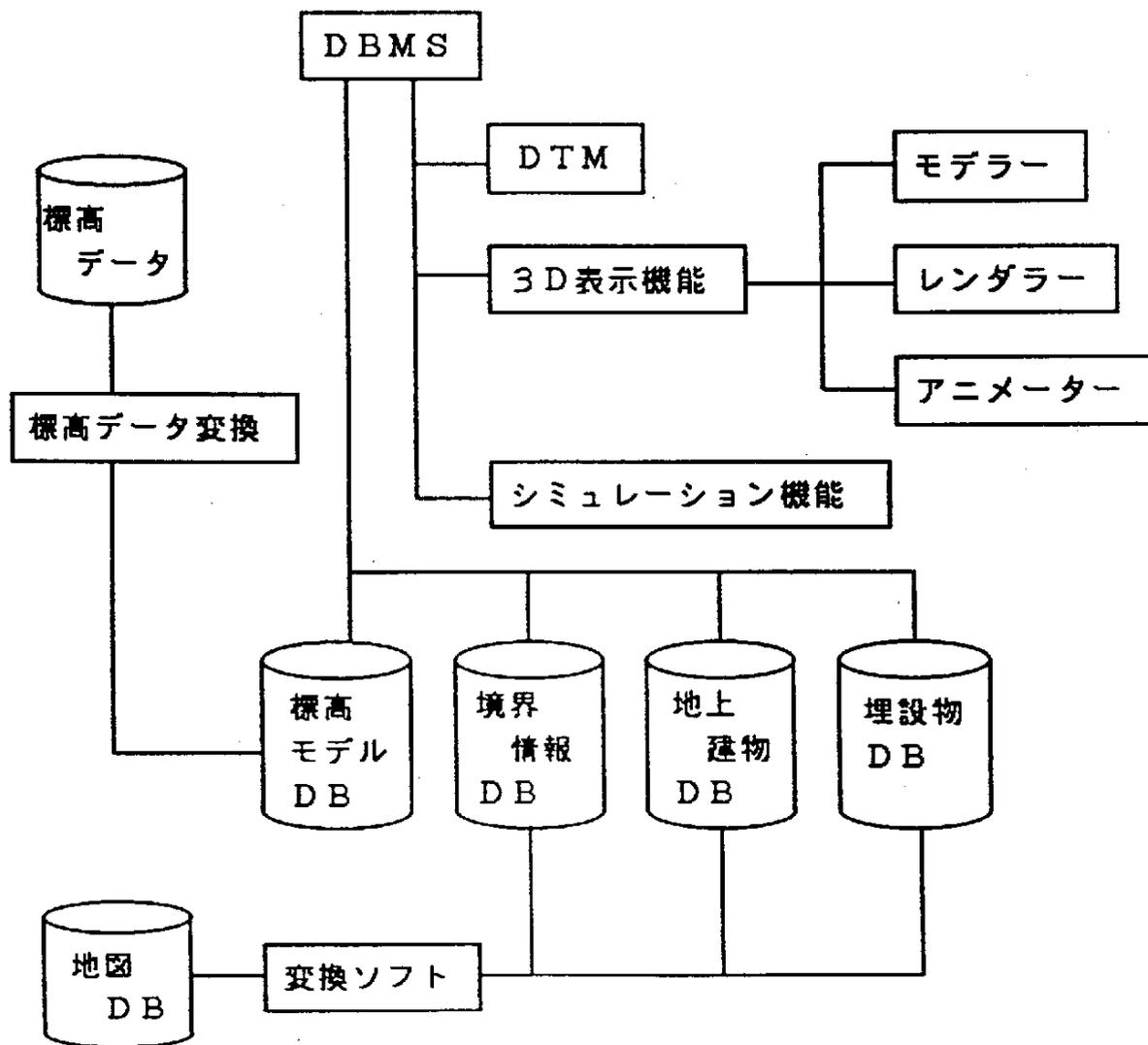


図4-3 データ構成の考え方

4. 3 出力機能の検討

本システムの出力機能としてはDTMとそのレンダリング結果の出図があるがその画質は利用場面に影響を与える。ここではDTMの表示手法を具体的に提示し、目的に即した表示を選択する。さらにこれらの表示形態が持つメリットについて整理する。

(1) DTM表示手法

DTMの表示手法のうちメッシュモデルは基本的なものである。しかし、メッシュモデルは内捜による任意性や等高線情報に忠実でないなどの難点があり、地形によっては十分その特徴を表現できないケースがある。ステップコンターモデルも同様にラフなモデルであるが例えば都市開発の初期イメージの評価などでは処理スピードの速さなどから使われることもある。

TINモデルは地形を滑らかに表示するのに有用であるが等高線情報に完全に忠実でなく、地域境界では不安定な振る舞いをしがちである。これに対してTINコンターモデルは比較的精度が高く、特に等高線に忠実であるため視覚的イメージと一致した地形が得られる。

実際の土木造成などのケースではある地形に対して一律に一つのDTMモデルを適用するのは無理でそれぞれの特性を活かして混合して使用するのが正しい。しかし本システムはこうした用途を想定していない。都市ユーティリティ事業者や自治体はむしろ使い勝手を優先させ標高データについては一律同一なDTMモデルでも実用になると考えられる。

これらの4モデルの関係を図4-4に示した。また表示イメージを図4-5に示す。

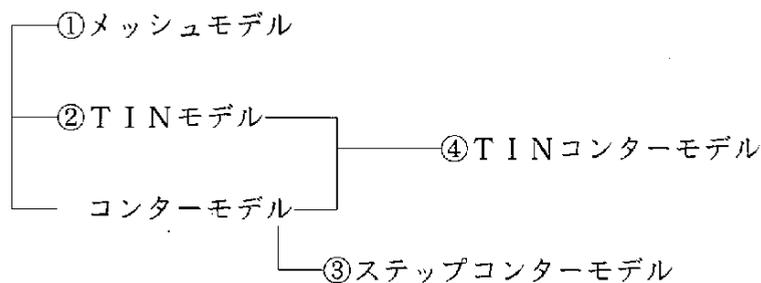
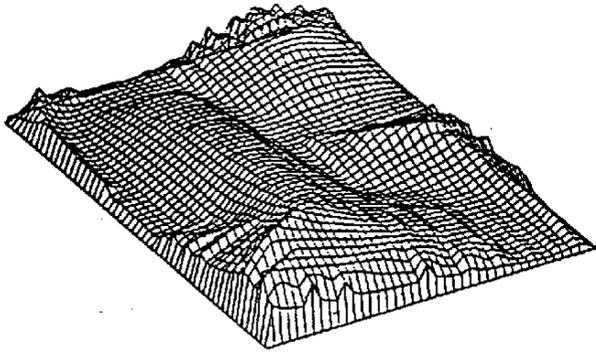
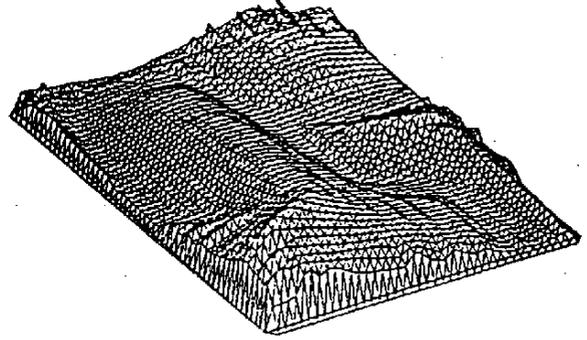


図4-4 表示タイプ

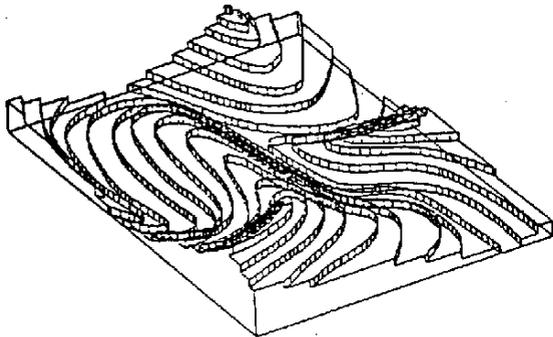
メッシュモデル



TINモデル



ステップコンターモデル



TINコンターモデル

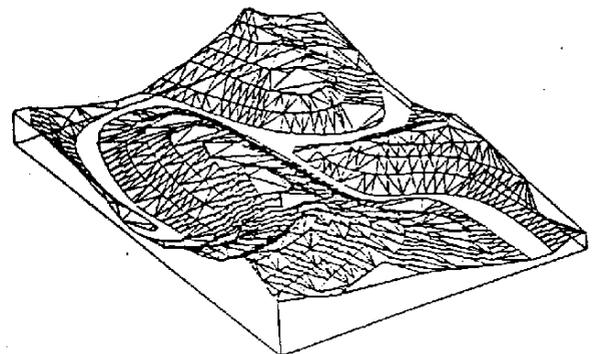


図 4 - 5 DTM表示の 4 タイプ

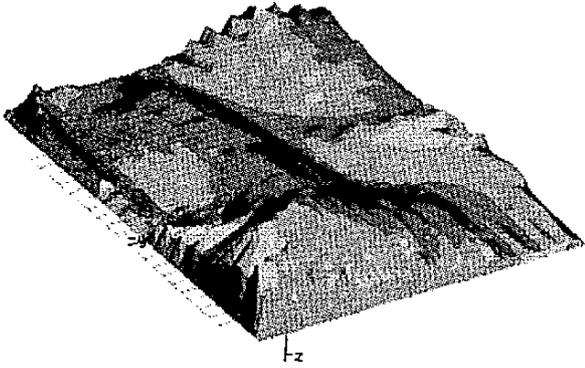
(2) DTM表示手法の評価

これらのメッシュ分割手法は他の関連アプリケーション、例えば景観の表示を行う際に異なる特性を示す。これを検査するために上記の標高モデルをベースにレンダリングを行った。図4-6はその結果をグレー表示したものである。

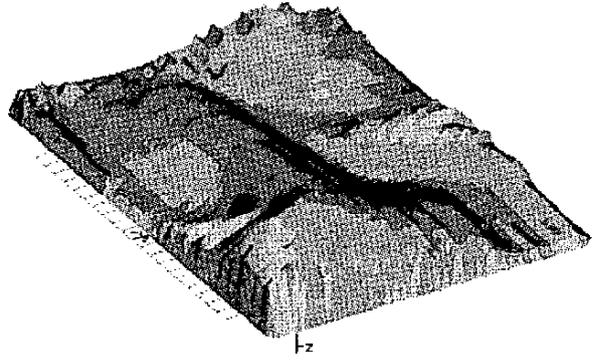
単純な格子状のメッシュモデルは地形境界付近でジャギイが現れている。同様にTINメッシュもジャギイが存在する。ステップモデルは等高線の忠実な表現であるが特殊な用途を除いては実用的とはいえない。TINコンターモデルはそのなかでも比較的自然的な表現になっている。

レンダリングによって比較的良質な画像を必要とする場合にはTINコンターモデルが適している。ただ、このモデルは建築物を同時に表示した場合に不可避な建築面の傾斜などを考えるとステップモデルも一部取り込んで合成できる機能が必要であろう。

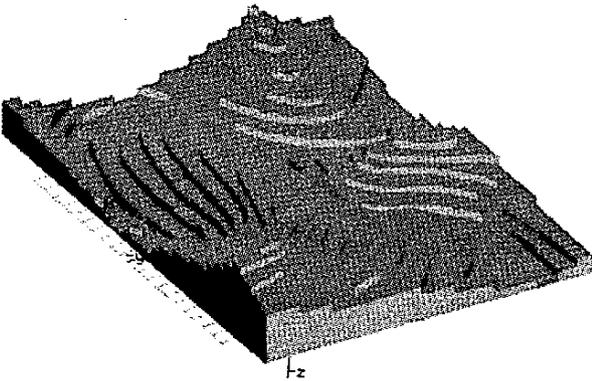
メッシュモデル



TINモデル



ステップコンターモデル



TINコンターモデル

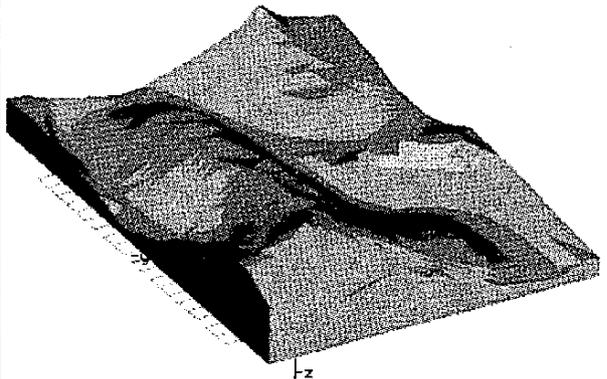


図 4 - 6 D T M 画像の 4 タイプ

(3) 3次元標高モデルの有効性

ベースマップの3次元化は3次元標高モデルの生成につながる。これをさらにレンダリングすることによって都市空間の可視性が強まる。その結果として地図では判断できなかった属性が直観的に理解できるものとなる。

これは例えば都市管理においても適合する。過密した大都市では輻輳化する埋設物を地表面との位置関係を表示できるため対象物を選択したり確認することが非常に困難である。しかし標高モデルの登場で管理が容易になると考えられる。

この標高モデルの特徴は広域性である。行政区画単位でこの情報を保持することによって公共性の高いデータベースを構築し、民間の多様なニーズに応えるべく公開することも可能であろう。

標高モデルは都市インフラ設備管理面では次の諸点で従来の地図データベースの面情報より優越していると考えられる。

- ①都市インフラ設備に関連している土木作業に対して有効な可視化情報を提供する。
- ②CAD機能を取り込んで設計、施工から管理までを一元化できる。
- ③地上から地下まで建造物の内部まで踏み込んだインフラ管理が実現できる。
- ④複合建築物、例えば人工地盤、軌道上ビルやスーパー堤防なども表現できる。
- ⑤ゼネコンやデベロッパー、建築事務所など他業種との情報交換がDXFファイル変換などで容易になる。同時に業務提携も促進される。

5. 成果のまとめ

5. 1 成果概要

今年度はシステムの利用局面の整理を踏まえてデータ設定、基幹機能構成を検討した。地図データベースに依拠する3次元データの作成と出力イメージについても検討を行い、実用システムの仕様、基本設計の基礎資料、基礎データを提供できるプロトタイプの構築を実施した。以上から次のような成果が得られた。

- ・3次元マップの一般的作製手法を確立した。これによりこれまでネックとなっていた3次元データの自動作成も将来的に可能である。
- ・DTM（デジタル標高モデル）と地図データベースの結合が有効であることを確認した。これは都市開発や都市計画支援での基礎情報である。
- ・都市管理システムとして設備管理などでの場面の実用性を検証できた。
- ・出力データに対して用途に応じた画質の標準ガイドラインを策定した。
- ・テストデータとして、川崎駅周辺部の建築物、鉄道、道路の概況、新宿副都心の地上・地下埋設物の3次元形状データを作製した。ファイル容量は0.5M、0.8M程度である。ベクターデータであるためこの程度になった。

こうした主要な成果の他にシステムのアウトラインが可視化され今後実用化システムを開発する際の基盤を提供できるようになった意義は大きいといえる。また、これまで散在していたシステム構築技術を統合できたことも無視できない効果であるといえる。

5. 2 今後の課題

既製システム、特にGISとのデータ交換を形状のみにとどまらず属性情報をも包含させることが望ましい。そのためにはSQL、informixといった汎用データベースの知識が求められるだけでなく、3次元表示中での物体の選択の方法やグラフィックデータの交換フォーマットについてのノウハウも必要となろう。さらにイメージデータの様々な加工編集機能が画像処理に付随し、ラスターベクターの変換技術もこの内に

ふくめれば一層広範囲なソフトウェア技術者の動員が望まれる。

いうまでもなく景観画面の作成はCG技術をフルに必要とする。CGソフトを構成する三つのモジュール、モデラー、レンダラー、アニメーターについても一からの開発に際しては各々専門技術が必要となる。ネットワーク構築もノウハウがあり、セキュリティ機能設計、異機種接続などが要求され、これらを統合するシステムインテグレーションが要求される。

こうした開発全体に関わる課題の他に下記のような項目が重点的に検討されるべき課題となる。

① 2D-3D描画機能

モデラー機能の充実。現在のプロトタイプでは外部DBまかせであるがこの機能は実用システムでは不可欠なものとなる。

② 地図データベース情報の変換機能の強化

地図データベースの保持するデータのタイプは「2.6」で整理したように4つのテーマに集約される。その中でもラスターデータやベクターデータのトポロジカル構造などは

地図データベースの最も特徴的なデータであるが標準的なフォーマットといえるものが少なく、また変換技術も高度なテクニックが要求される。実用システムではこれを解決して

おかねばならないのはもちろんであるが3次元空間でこうした情報をどのように表現し使うかということも大きな問題となることも事実である。

③ データ容量の問題

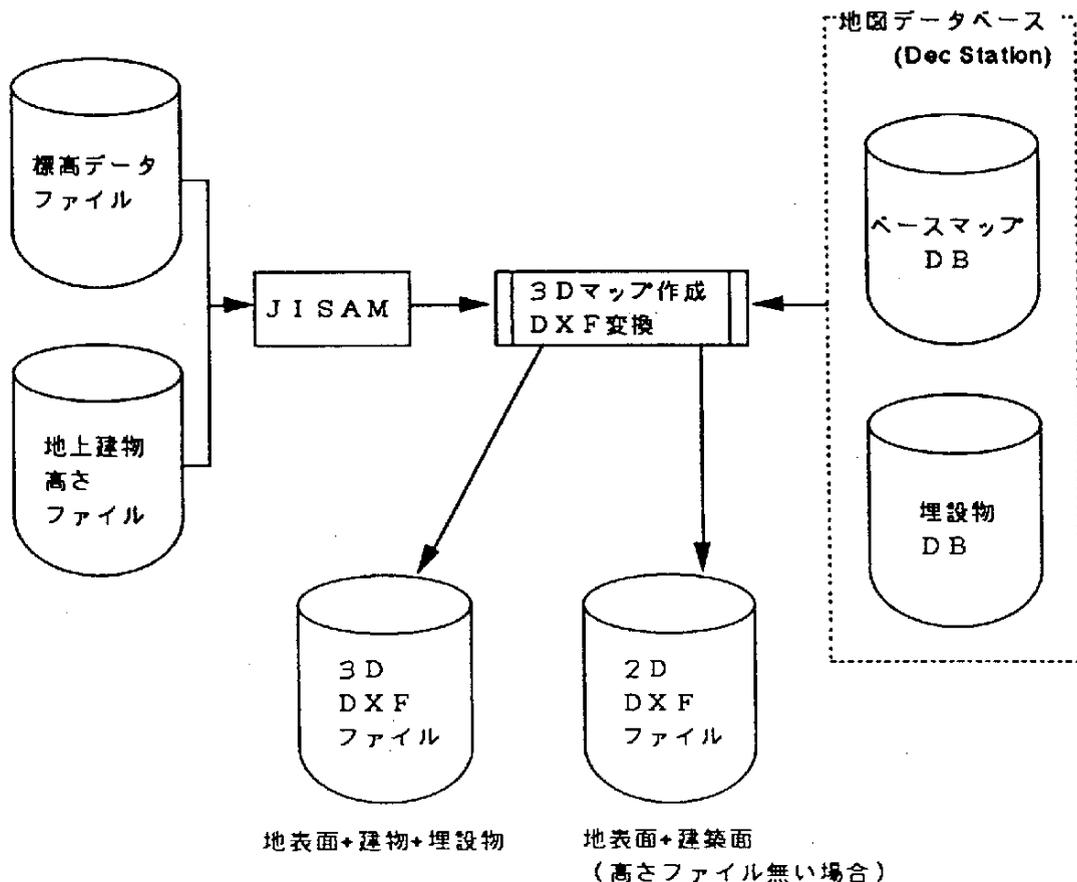
上記のように地図データベースの保持するラスターデータを実用化システムでは取り込むことが必要になる。この時、避けて通れないのがデータ容量の問題である。管理対象地域が広大であればあるほどデータは膨大なものとなりうる。特にリモートセンシングや航空写真などのイメージデータは都市計画などでは不可欠でもあり、JPEGなどの画像圧縮方式の採用によってこれを圧縮することが求められるであろう。

補足資料1 DXFファイル変換プログラムについて

地図データベースの保持するベースマップや都市ユーティリティ（ガス、水道など）のデータはそのままでは3次元化できない。最も汎用性のある方法はここで述べるDXFファイルに地図データの保有するベクターデータ、例えば行政区ポリゴン、道路、鉄道、河川などを変換する手法である。これによって広く普及している地図データベース情報を3次元化することが可能になる。高さ情報はJISAMによって付加できる。

ファイル変換の考え方は下記のように図式化される。ここでは都市管理などの局面でも利用できるようインフラDBを想定して設計した。高さデータが入手できない地域があることを想定して2次元のDXFファイルにも対応している。

以下そのプログラム仕様を示す。



付図1-1 システムフローチャート

| | | | ファイル名 |
|--------|-------------|-------|------------|
| | | | DXFCOM. h |
| ヘッダー名 | 変数名 | 属性 | 説明 |
| DXFCOM | IMUNO (100) | I * 4 | 地図番号テーブル |
| | IMUSU | I * 4 | 地図枚数 |
| | IOVNO (100) | I * 4 | オーバーレイテーブル |
| | IOVSU | I * 4 | オーバーレイ数 |
| | IX | I * 4 | 最小X座標 |
| | IY | I * 4 | 最小Y座標 |
| | IRUND | I * 4 | ポリゴンラウンド値 |
| | IFLAG | I * 4 | 反転フラグ |
| | ICUTR | I * 4 | 小図形カット範囲 |
| | IMINX | I * 4 | 地図左下のX座標 |
| | IMINY | I * 4 | 地図左下のY座標 |
| | IMAXX | I * 4 | 地図右上のX座標 |
| | IMAXY | I * 4 | 地図右上のY座標 |
| | | | |
| | | | |
| | | | |
| | | | |

| | |
|-------|-----------|
| ファイル名 | LOGCOM. h |
|-------|-----------|

| ヘッダー名 | 変数名 | 属性 | 説明 |
|--------|---------|-------|----------------|
| DXFCOM | INUMBER | I * 4 | レコード数 |
| | ILINDT | I * 4 | 2点しかもたないレコード数 |
| | IPLYDT | I * 4 | ポリゴンにならないレコード数 |
| | ISMLDT | I * 4 | 小図形レコード数 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | |
|-------|-----------|
| ファイル名 | MAPDXF. h |
|-------|-----------|

| ヘッダー名 | 変数名 | 属性 | 説明 |
|--------|-------------------------------|----|----|
| MAPDXF | MAPCOM(MAXSIZ) | | |
| | MAPCNT(256) | | |
| | | | |
| | | | |
| | | | |
| | PARAMETER (MAXSIZ = 400000) | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|-------------------|----|---|---------|
| ファイル名 | DXF_MAIN.C | 言語 | C | FUN・SUB |
| 機能 | DXFファイル作成メインプログラム | | | |
| 呼出し | | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-----|--------|----|----|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|------------------|----|---|---------|
| ファイル名 | DXF_BLK. C | 言語 | C | FUN・SUB |
| 機能 | ブロックセッションの作成を行う。 | | | |
| 呼出し | DXF_BLK (IUNIT) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-------|--------|----|----------------|
| IUNIT | I * 4 | I | DXFファイルのユニット番号 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|--|----|---|---------|
| ファイル名 | DXF_FLRD. C | 言語 | C | FUN・SUB |
| 機能 | ISAMファイルを読み込み高さとおオーバーレイ番号をセットする。 | | | |
| 呼出し | DXF_FLRD (IUNIT, IDATA, ITAKASA, ILAYER, IRET) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|---------|--------|----|-----------------|
| IUNIT | I * 4 | I | ISAMファイルのユニット番号 |
| IDATA | I * 4 | I | 家形のデータ |
| ITAKASA | I * 4 | O | 高さデータ |
| ILAYER | I * 4 | O | オーバーレイ番号 |
| IRET | I * 4 | O | リターンステータス |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|------------------|----|---|---------|
| ファイル名 | DXF_HED. C | 言語 | C | FUN・SUB |
| 機能 | ヘッダーセクションの作成を行う。 | | | |
| 呼出し | DXF_HED (IUNIT) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-------|--------|----|----------------|
| IUNIT | I * 4 | I | DXFファイルのユニット番号 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|---|----|---|---------|
| ファイル名 | DXF_MUSET. C | 言語 | C | FUN・SUB |
| 機能 | 地図番号をセットする。 | | | |
| 呼出し | DXF_MUSET (IMINMU, IMAXMU, ITBMAX, IMUY, IMUX, IMODE, IRET) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|------------|--------|----|----------------|
| IMINMU | I * 4 | I | 入力した最終地図番号 |
| IMAXMU | I * 4 | I | 入力した最大地図番号 |
| ITBMAX (*) | I * 4 | O | 地図番号テーブル |
| IMUY | I * 4 | O | 地図番号テーブルセット数 Y |
| IMUX | I * 4 | O | 地図番号テーブルセット数 X |
| IMODE | I * 4 | I | パラメータセレクトモード |
| IRET | I * 4 | O | リターンステータス |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|------------|----|---|---------|
| ファイル名 | DXF_RND. C | 言語 | C | FUN・SUB |
|-------|------------|----|---|---------|

| | |
|-----|------------------------------|
| 機能 | ポリゴン図形のチェックを行う。 |
| 呼出し | DXF_RND (IXY, IMAX, IRNDFLG) |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-------------|--------|-----|----------------|
| IXY (2,200) | I * 4 | I/O | 座標テーブル |
| IMAX | I * 4 | I/O | 座標数 |
| IRNDFLG | I * 4 | O | ラウンドフラグ =1: 利か |
| | | | =2: 利かではない |
| | | | =3: 小図形 |
| | | | |
| | | | |

| | | | |
|-----------|---------|------|-------|
| サブルーチン説明書 | システム名 | PAGE | 1 / 1 |
| | 3次元マップ化 | DATE | |

| | | | | |
|-------|---------------------|----|---|---------|
| ファイル名 | DXF_VEC. C | 言語 | C | FUN・SUB |
| 機能 | ベクター方向の決定を行う。 | | | |
| 呼出し | DXF_VEC (IPSU, IXY) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-------------|--------|-----|--------|
| IPSU | I * 4 | I | 座標数 |
| IXY (2,200) | I * 4 | I/O | X, Y座標 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|------------------|----|---|---------|
| ファイル名 | DXF_TBL. C | 言語 | C | FUN・SUB |
| 機能 | テーブルセクションの作成を行う。 | | | |
| 呼出し | DXF_TBL (IUNIT) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-------|--------|----|----------------|
| IUNIT | I * 4 | I | DXFファイルのユニット番号 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|--------------------------------|----|---|---------|
| ファイル名 | DXF_OV12.C | 言語 | C | FUN・SUB |
| 機能 | 街区のエンティティセクションを行う。 | | | |
| 呼出し | DXF_OV12 (IUNIT1, IDATA, IRET) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-----------|--------|----|----------------|
| IUNIT | I * 4 | I | DXFファイルのユニット番号 |
| IDATA (*) | I * 4 | I | 街区データ |
| IRET | I * 4 | I | リターンステータス |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|--|----|---|---------|
| ファイル名 | DXF_OV27.C | 言語 | C | FUN・SUB |
| 機能 | 家形のエンティティセクション作成を行う。 | | | |
| 呼出し | DXF_OV27 (IUNIT1, IUNIT2, IDATA, IRET) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-----------|--------|----|-----------------|
| IUNIT1 | I * 4 | I | DXFファイルのユニット番号 |
| IUNIT2 | I * 4 | I | ISAMファイルのユニット番号 |
| IDATA (*) | I * 4 | I | 家形データ |
| IRET | I * 4 | O | リターンステータス |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|--------------------------------|----|---|---------|
| ファイル名 | DXF_ENT. C | 言語 | C | FUN・SUB |
| 機能 | DBを読み込み、エンティティセクションの作成を行う。 | | | |
| 呼出し | DXF_ENT (IUNIT1, IUNIT2, IRET) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|--------|--------|----|-----------------|
| IUNIT1 | I * 4 | I | DXFファイルのユニット番号 |
| IUNIT2 | I * 4 | I | ISAMファイルのユニット番号 |
| IRET | I * 4 | O | リターンステータス |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|-----------------------|----|---|---------|
| ファイル名 | DXF_OV2.C | 言語 | C | FUN・SUB |
| 機能 | 埋設物のエンティティセクション作成を行う。 | | | |
| 呼出し | DXF_OV2 (IUNIT, IST) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|-------|--------|----|--------------------|
| IUNIT | I * 4 | I | DXFファイルのユニット番号 |
| IST | I * 4 | O | リターンステータス = 0 : 正常 |
| | | | ≠ 0 : エラー |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|--|----|---|---------|
| ファイル名 | DXF_OV2_1.C | 言語 | C | FUN・SUB |
| 機能 | 埋設物の引出し線情報のエンティティセクション作成を行う。 | | | |
| 呼出し | DXF_OV2_1 (IUNIT, IDATA, PSX, PSY, PSZ, PEX, PEY, PEZ, ITBLSU1, IDATASU, IRET) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|----------------|--------|----|----------------|
| IUNIT | I * 4 | I | DXFファイルのユニット番号 |
| IDATA (*) | I * 4 | I | 埋設物データ |
| PSX (1000,100) | R * 4 | I | 埋設物始点X座標 |
| PSY (1000,100) | R * 4 | I | 埋設物始点Y座標 |
| PSZ (1000,100) | R * 4 | I | 埋設物始点Z座標 |
| PEX (1000,100) | R * 4 | I | 埋設物終点X座標 |
| PEY (1000,100) | R * 4 | I | 埋設物終点Y座標 |
| PEZ | R * 4 | I | 埋設物終点Z座標 |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|-------------|----|---|---------|
| ファイル名 | DXF_OV2_1.C | 言語 | C | FUN・SUB |
|-------|-------------|----|---|---------|

| | |
|-----|--|
| 機能 | 埋設物の引出し線情報のエンティティセクション作成を行う。 |
| 呼出し | DXF_OV2_1 (IUNIT, IDATA, PSX, PSY, PSZ, PEX, PEY, PEZ, ITBLSU1, IDATASU, IRET) |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|------------|--------|----|---------------------------------|
| ITBLSU1 | R*4 | I | 埋設物座標のレコード数 |
| IDATASU(*) | R*4 | I | 埋設物座標のデータ数 |
| IRET | R*4 | O | リターンステータス = 0 : 正常 ≠ 0 : エラー |
| | | | |
| | | | |
| | | | |
| | | | |

| | | | |
|-----------|------------------|------|-------|
| サブルーチン説明書 | システム名 3次元マップ化 | PAGE | 1 / 1 |
| | | DATE | |

| | | | | |
|-------|--|----|---|---------|
| ファイル名 | DXF_CHK. C | 言語 | C | FUN・SUB |
| 機能 | ある一点がライン上にあるかの判定を行う。 | | | |
| 呼出し | DXF_CHK (RX1, RY1, RX2, RY2, RDX, RDY, ICHK) | | | |

| 変数名 | タイプ・長さ | 入出 | 説明 |
|------|--------|----|------------------|
| RX1 | R*4 | I | 線分の始点X座標 |
| RY1 | R*4 | I | 線分の始点Y座標 |
| RX2 | R*4 | I | 線分の終点X座標 |
| RY2 | R*4 | I | 線分の終点Y座標 |
| RDX | R*4 | I | 判定される点のX座標 |
| RDY | R*4 | I | 判定される点のY座標 |
| ICHK | R*4 | O | 判定結果 = 0 : オンライン |
| | | | = -1 : ライン外 |

補足資料 2 J I S A Mライブラリについて

1. 概要

J I S A Mライブラリは、データベース分野で開発された I S A M (Indexed Sequential Access Method) とよばれる技法を各種アプリケーションで利用するためのツールである。この J I S A Mは、地図データベースに含まれるオブジェクトに高さ情報を付加するための強力な汎用ツールでもある。

通例では I S A M機能は複数の基準のうちの1つを基にレコードを選択しなければならないような複雑なファイルの管理に適している。そのような性質上から、利用ユーザは少なくとも VAX FORTRAN や、MS-BASICなど多くの高級言語で標準装備されている。

ところが、DEC station の標準OSである U L T R I X上には、I S A Mを利用できる環境が無いというのが現状である。

U L T R I Xは、U N I Xの派生OSであることから、UL-JISAMはU N I Xと親和性の高いC言語で開発を行った。

当然のことながら、UL-JISAMを使用するにあっては、C言語開発のプログラムとのリンクが一番好ましいのであるが、一部の機能を除いてFORTRAN でもリンク・使用できるようにした。

2. U L - J I S A Mとは

2. 1 データ・ファイルと索引ファイル

大量のデータを扱うにはデータを効率よく管理しなければならない。そのため U L - J I S A M では、固定長データ・ファイルと索引ファイルの2種類のファイルで構成される。データ・ファイルに記録されるデータ(データ・レコード)は付図2-1に例を示すように、いくつかのフィールドで構成される。索引ファイルは目的のデータを迅速に検索するための索引(インデックス)を管理するファイルである。付図2-2に索引ファイルとデータ・ファイルとの関係を示す。

ある社員番号をもつ社員データを読み出すという操作を考えればわかるように、デ

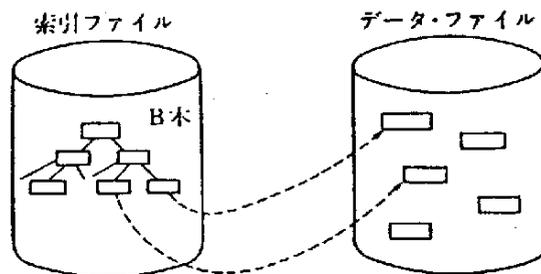
ータを読み出すには一般に、指定フィールドが指定された値（キー値）であるデータを検索する機能が必要である。索引はこうした検索を効率よく行うためのもので、
 <キー値, データの位置>という対をキー値の順序で管理している。索引をキー値で検索すればそれに対応したデータの位置が得られるから、それによってデータを読み出す。このような検索を（キーに関する）ランダム検索という。

また、たとえば年齢や社員名の順序で検索するなど、キー値の順序で検索したいことがある。これを（キーに関する）順次検索という。

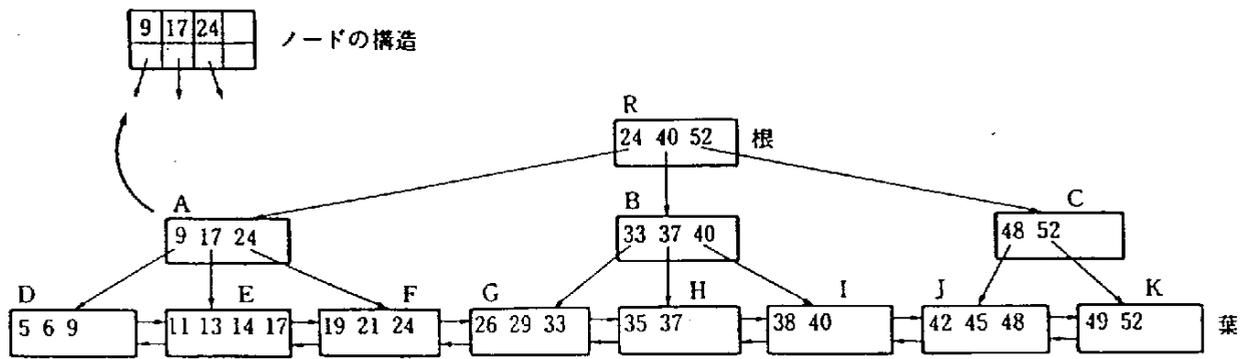
キーに関するこの2種類の検索（アクセス）方法をISAM（検索引つき順次アクセス法）とよぶ。検索を行うためには、キー値に順序（大小関係）が定義されている必要がある。

| 社員番号 | 社員名 | 出生年 |
|------|-------|------|
| 19 | 宮沢 りえ | 1973 |

付図 2-1 データ・レコード



付図 2-2 索引ファイルとデータ・ファイルの関係



付図 2 - 3 B木の例

2. 2 B木の構造

UL-JISAMの索引はB木という手法を用いて実現する。B木はデータベースの研究者によって考え出されたもので、検索はもちろんのこと、動的な登録、削除が効率よく行えるので、データベース管理システムの重要な実現技術のひとつになっている。

B木の例を付図-3に示す。これは一般にB+木とよばれているものがある。

親ノードをもたないノードを根、子ノードをもたないノードを葉という。ノードには<キー値, 子リンク>という対が記録される。この対を以下では要素とよぶことにする。子リンクというのは子ノードの位置を示す値（ファイル先頭からのオフセット）である。ノードNの親ノード中の要素で、子リンクがノードNを指している要素をノードNの親要素という。ノード、要素のどちらを意味しているのか前後から明白である場合はたんに親ということもある。

索引ファイルのアクセス単位はノードである。

ノード内の要素はキー値の昇順で記録されている。葉にはすべてのキー値が出そろっていて、昇順にならんでいる。順次検索を効率よく行うため、葉は左右の葉へのポインタ（左、右リンク）をもっている。また、葉以外のノードに関しては、要素のキー値は、その子ノードの最大のキー値（つまり子ノードの右端要素のキー値）に一致させている。いいかえると、親要素のキー値は子ノードの最大キー値に一致している。したがって、根の最大キー値はそのB木の最大キー値となっている。

葉の要素は子ノードをもっていない。そこでUL-JISAMでは、葉の要素の子リンクの欄に、データ・レコードの位置（その要素のキー値に対応するデータのデータ・ファイル上の位置）を入れることにした。

キーには、生年月日のように重複して出現する場合と、社員番号のように唯一の場合がある。これをそれぞれ重複可能キー、唯一キーとよぶことにする。重複可能キーの場合、同一キーが存在するときは要素をデータ・レコードの位置の昇順に記録している。

3. UL-JISAM使用前に

3.1 VAX FORTRANのISAMとUL-JISAMの相違点

UL-JISAMは、C言語で開発していること等から、VAX FORTRAN のISAMとは制御体系がかなり異なる。そういう意味からすればVAX FORTRAN のエキスパートにとって UL-JISAM の印象は、非常に扱いにくいライブラリと映るであろう。そのような向きには、UL-JISAMはVAX FORTRAN のISAMとは全く異なるものと割り切って使用していただくことを勧める。

主な、VAX FORTRAN のISAMとUL-JISAMの違いを以下に記す。

- ①制御体系が全く異なる。
- ②VAX FORTRAN は、データと索引が一つのファイルであったが、UL-JISAMはデータ・ファイルと索引ファイル（キーの数だけファイル数がある）とで構成される。
- ③VAX FORTRAN は、一つのレコードを制御するにあたり、READ文又はWRITE 文で制御できたがUL-JISAMは、上記の通り複数のデータで構成されるためデータファイル、索引ファイルのおのおのをプログラムレベルで管理する必要がある。
- ④VAX FORTRAN で作成したISAMファイルは、順次アクセスも可能であったがUL-JISAM は索引ファイルを用いて順次検索を行う。

3. 2 UL-I S A Mの動作環境と提供ファイル構成

UL-JISAMの動作可能な環境は、ハードがDEC station で、OSがULTRIXである。使用可能な高級言語は、DEC C とFORTRAN 77である。

ソフトウェアとしてのUL-JISAMの構成ファイルは、以下の2ファイルである。

isam.a : UL-JISAMのライブラリ

isamh.h : UL-JISAMのヘッダファイル (DEC Cのみ使用)

3. 3 UL-I S A Mライブラリの概要

UL-JISAMのデータファイルは固定長のみ扱える。管理するファイルはデータ・ファイルと索引ファイルの二つである。

索引ファイルでは1個の索引 (B木) を管理する。キーの型は、整数、浮動小数点数、文字列が扱える。また、キーは索引ごとに重複の可否を指定できる。

ライブラリ関数群を次葉の表-1に示す。

付表2-1 ライブラリ関数群

<データ・ファイル> (固定長レコード)

| | |
|-----------------|---------------------------|
| dbf__create__() | データ・ファイルを新規作成する |
| dbf__open__() | データ・ファイルをオープンする(FORTRAN用) |
| dbf__open() | データ・ファイルをオープンする(C言語用) |
| dbf__close__() | データ・ファイルをクローズする |
| dbf__flush__() | バッファの内容をフラッシュする |
| dbf__read__() | レコードを読み出す |
| dbf__regist__() | レコードを登録する |
| dbf__delete__() | レコードを削除する |
| dbf__update__() | レコードを書き出す(更新用) |

dbf__regno__() 次の登録番号を取り出す
dbf__recs__() 現在の登録レコード数を取り出す
dbf__dump__() データ・ファイルの内容をダンプする

<索引ファイル>

idx__create__() 索引ファイルを新規作成する(FORTRAN用)
idx__create() 索引ファイルを新規作成する(C言語用)
idx__mktype__() 文字列の索引ファイルを作成する際にキータイプを作成
idx__open__() 索引ファイルをオープンする(FORTRAN用)
idx__open() 索引ファイルをオープンする(C言語用)
idx__close__() 索引ファイルをクローズする
idx__flush__() バッファの内容をフラッシュする
idx__elems__() 登録要素数を取り出す
idx__top__() 最初の要素を検索する
idx__bottom__() 最後の要素を検索する
idx__next__() 次の要素に進む
idx__prev__() 前の要素に戻る
idx__stat__() 現在のカーソルの状態を取り出す
idx__curr_record__() 現在の要素が指すデータレコードの位置を取り出す
idx__search__first__() 指定キーの最初の出現を検索する
idx__search__next__() 指定キーの次の出現を検索する
idx__set__sinf() AND検索用の構造体sinfを初期設定する*
idx__msearch__first() 指定キーの最初の出現を検索する(AND検索)*
idx__msearch__next() 指定キーの次の出現を検索する(AND検索)*
idx__search__ge__() 指定キー以上のキーを持つ最初の出現を検索する
idx__seek__() 要素を検索する
idx__regist__() 要素を登録する
idx__delete__() 現在の要素を削除する
idx__del__first__() 指定キーをもつ最初の要素を削除する
idx__del__() 要素を指定して削除する

idx_dump__() 索引ファイルの内容をダンプする

注) *は、C言語のみ使用可

4. プログラミング使用説明

UL-JISAMは、C言語プログラムとFORTRAN 77プログラムとリンクが可能である。プログラム時の関数使用手順は以下の付図2-4のとおりになる。(関数の引き数は省略)

a. ISAM関連ファイル生成

ファイル生成

```
dbf_create__();
```

```
idx_mktype__();
```

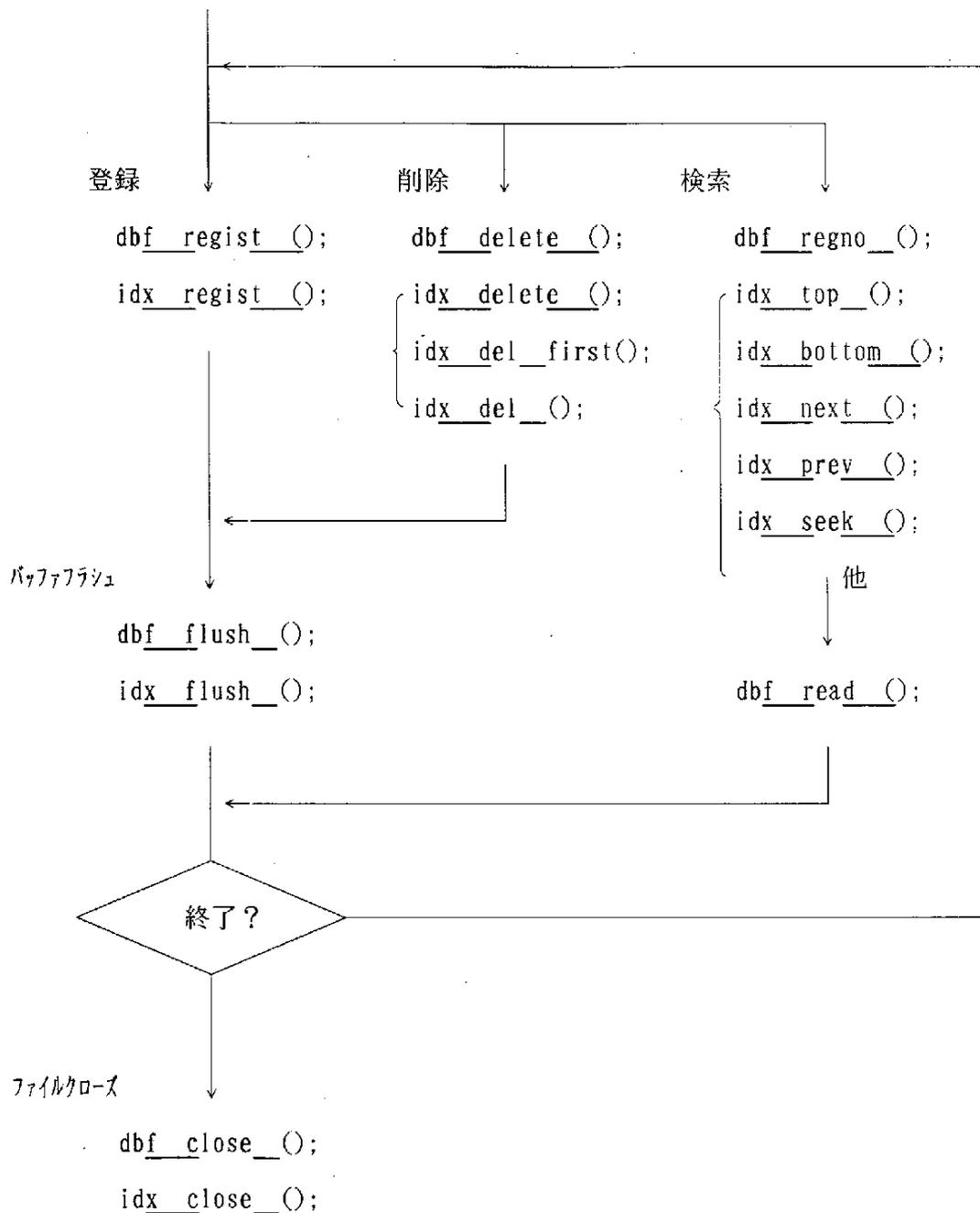
```
idx_create();
```

b. ISAM関連ファイル制御

ファイルオープン

dbf_open();

idx_open();



付図2-4 ISAMライブラリ関数の使用フロー

5. UL-JISAM関数

5. 1 データ・ファイルの新規作成

◇void dbf_create_(char *filename, short rec_size)

データ・ファイルの作成と初期化を行う関数であり、最初に一度だけ呼び出す。引数としてファイル名とデータ長を指定する。同一名のファイルが存在する場合は、その内容が破壊される。

処理は、ファイルを作成して、ヘッダに管理情報の初期値を書き込むだけである。

5. 2 データ・ファイルのオープン、クローズ

◇long dbf_open_(DBF *dbfp, char *filename)

データ・ファイルをオープンする関数 (FORTRAN 用)。データ・ファイルをアクセスする前にこの関数を用いてオープンする。オープンしたデータ・ファイルの鑑紙情報へのポインタが返されるので、以降はそれを用いてアクセスを行う。オープンに失敗した場合は0を返す。処理は、管理情報の領域を確保し、ファイルをオープンしてヘッダに保存してある管理情報を読み出し、最後に、実行時の管理情報を設定する。

◇DBF *dbf_open(char *filename)

データ・ファイルをオープンする関数 (C言語用)。データ・ファイルをアクセスする前にこの関数を用いてオープンする。オープンしたデータ・ファイルの鑑紙情報へのポインタが返されるので、以降はそれを用いてアクセスを行う。オープンに失敗した場合はCの標準関数fopen()と同じく、NULLを返す。処理は、管理情報の領域を確保し、ファイルをオープンしてヘッダに保存してある管理情報を読み出し、最後に、実行時の管理情報を設定する。

◇int dbf_close_(DBF *dbfp)

データ・ファイルをクローズする関数。システムを終了するときには、この関数

を用いてデータ・ファイルをクローズする。Cの標準関数fclose()と同じく、成功の場合0, 失敗の場合-1を返す。

◇void dbf__flush__(DBF *dbf(P)

バッファの内容をフラッシュする関数。これは、登録や削除を行ったあと、ファイルにその変更を反映するのに用いる。

5. 3 データ・ファイルのアクセス

◇void dbf__read__(DBF *dbfp, void *bu(f)

登録されているデータを読み込む関数。loc でデータ・レコード位置を指定する。buf にはデータを読み込むレコード・バッファへのポインタを指定する。

◇long dbf__regist__(DBF *dbfp, void *bu(f)

データを登録する関数。bufで指定されたレコード・バッファの内容を書き出し、登録番号を増加する。返す値はデータを登録した位置である。この値は索引キー(要素)を登録するのに用いる。

◇void dbf__update__(DBF *dbfp, LOC *loc, void *bu(f)

データを書き出す関数だが、これはデータの更新を行うためのものである。loc でデータ・レコードを書き出す位置を指定する。dbf__regist__()とは異なり、レコード領域を確保することはしない。また、登録番号も増加しない。

◇void dbf__delete__(DBF *dbfp, LOC *loc)

データ・レコードを削除する関数で、指定されたレコード領域を削除リストの先頭につなぐ。loc には削除するデータ・レコードの位置を指定する。

◇long dbf__regno__(DBF *dbf(P)

次の登録番号を取り出す関数。データ・レコードに投入順序に関する情報を記録する場合に使用する。

◇long dbf__recs__(DBF *dbf(P))

登録レコード数を取り出す関数。これによってデータ・ファイルが空でないか調べられる。

5. 4 データ・ファイルのダンプ

◇void dbf__dump__(DBF *dbfp, int *leve(1))

データ・ファイルをダンプする関数。出力形式は2種類あり、levelで指定する。

level=0:ヘッダ, 削除リスト

level=1:ヘッダ, 削除リスト, 有効レコード

最初のほうでさまざまなチェックを行っているが、これはデバッグのためである。削除テーブルは、有効レコードのみを表示するのに使用する。削除レコードについては、その位置の一覧を示すだけで、内容は表示しない。

なお、内容表示はCRTになされる。情報としては、以下のものが出力される。

→fname:ファイル名

これはダンプのときにファイル名を表示するのに、またファイル・アクセスでエラーが生じたときのメッセージに使用する。ファイル名はとくに制限はない。

→wsize:作業領域の大きさ

データ・ファイルの場合は管理情報の大きさsizeof(DBF)である。メモ・ファイルと索引ファイルでは、ファイルのアクセスのためバッファを使用するので、その大きさが加算される。これはダンプ表示でのみ参照できる。

→rsize:レコード表

そのデータ・ファイルで管理されるデータ・レコードの長さで、レコード領域の確保、データのアクセスに必要な情報である。

ftype からここまではデータ・ファイルを新規作成したときにきまり、以降変化しない。

→recs:登録レコード数

登録されているデータ・レコードの個数。データの登録で増加し、削除で減少する。

→dels:削除レコード数

削除されて未使用となっているレコード領域の個数。削除で増加し、登録で減少する。

→eof:ファイルの最後

ファイルとして現在確保されている領域の最後の位置に1を加えた値を保持する。削除リストが空の場合、新しいレコード領域はこの位置から確保され、eofはレコード長だけ増やされる。

→del:削除リストの先頭

削除されたデータ・レコードは再利用するためにリストしているが、これはその先頭レコードの位置を保持するものである。削除レコードがない場合はLOC=NILである。

→regno:登録番号

次に登録されるデータの登録番号。初期値は1で、データを登録すると1増える。登録番号をデータ・レコードに設定するのはアプリケーションの責任である。すなわち、def_regno__()で登録番号を読み出し、データ・レコードに設定する。

5. 5 索引ファイルの新規作成

◇void idx_create__(char *filename, ulong *keytype, char *unique)

索引ファイルを新規作成する関数（FORTRAN用）。キーごとに最初に一度だけ呼び出し、そのキーに対する索引ファイルの作成と初期化を行う。引数として、ファイル名、キーの型重複登録の可否を指定する。

◇void idx_create(char *filename, ushort keytype, char uniqu (e)

索引ファイルを新規作成する関数（C言語用）。キーごとに最初に一度だけ呼び出し、そのキーに対する索引ファイルの作成と初期化を行う。引数として、ファイル名、キーの型重複登録の可否を指定する。

◇unsigned short idx_mktype__(unsigned short * le(n)

idx_create__または、idx_createで文字列の索引ファイルを作成する際に用いる。戻り値は、キータイプ（keytype）である。

5. 6 索引ファイルのオープン、クローズ

◇long idx_open__(IDX *idxp, char *filename(e) : FORTRAN用

◇IDX*idx_open(char *filename (e) : C言語用

◇int idx_close__(IDX *idx(p)

◇void idx_flush__(IDX *idx(p)

それぞれ、索引ファイルをオープンする関数、索引ファイルをクローズする関数、バッファの内容をフラッシュする関数。

◇long idx_elems__(IDX *idx(p)

登録要素を取り出す関数。

5. 7 順次検索

順次検索には、現在の要素の位置を記憶するためのカーソルを用いる。ここでは現在のノード(現在の要素を含むノード)が正規バッファに存在するように制御することにする。

以下の順次検索関数を用意する。

◇long idx_top__(IDX *idx(p)

◇long idx_bottom__(IDX *idx(p)

◇long idx_next_(IDX *idx(P))

◇long idx_prev_(IDX *idx(P))

カーソルをそれぞれ、最小のキーをもつ要素、最大のキーをもつ要素、前の要素、次の要素に移動する関数である。適切な要素に移動できた場合にはそれに対応したデータ・レコードの位置を返し、失敗した場合は-1を返す。

◇int idx_stat_(IDX *idx(P))

◇long idx_curr_record_(IDX *idx(P))

それぞれ、カーソルの状態、現在の要素のキーへのポインタ、現在の要素が指すデータ・レコードの位置を返す関数です。

5. 8 ランダム検索

◇long search_ge_(void *key(P))

指定キー値以上のキーのうちで最小のキーをもつ最初の要素を検索する関数。EOF位置になった場合は1、それ以外は0を返す。検索した位置はカーソルに設定されている。

検索に成功すると、検索した要素に対応するデータ・レコードへの位置を返し、検索に失敗したときは-1を返す。

◇long idx_search_first_(IDX *idxp, void *key(P))

◇long idx_search_next_(IDX *idxp, void *key(P))

前者は、指定キーの最初の出現を検索する関数。search_ge_()によって指定キー以上のキーをもつ最初の要素に移動し、そのキーの登録時に登録位置をきめるのにも用いる。後者は、指定キーの次の出現を検索する関数。

◇long idx_search_ge_(IDX *idxp, void *key(P))

指定キー以上の最初の出現を検索する関数。

さらに、登録位置の検索や、削除する要素のため、キーとデータ・レコードの位置を指定して要素を検索する関数idxseek()を用意する。この関数は、要素を登録す

る関数 `idx_regist_()` と、要素を指定して削除する関数 `idx_del_()` で使用する。

◇ `int idx_seek (IDX *idxp, void *keyp, LOC *loc)`

要素を検索する関数で、検索できたときは0、検索できなかったときは1を返す。

5. 9 複数キーの値のAND検索

AND検索のためのライブラリ関数は以下の三つです。

注) AND検索は、C言語のみ使用可

◇ `void idx_set_sinf (IDX *idxp, void *keyp, SINF *sinf)`

検索する索引ファイルの管理情報へのポインタと検索キーへのポインタを `*sinf` に設定する。

◇ `long idx_msearch_first (int n, SINF *sinf)`

◇ `long idx_msearch_next (int n, SINF *sinf)`

それぞれ、AND条件をみたす最初の出現を検索する関数、AND条件をみたす次の出現を検索する関数。

アプリケーションでは、構造体 `SINF` の配列を用意し `idx_set_sinf()` で索引ファイルと検索キーへのポインタを設定して、`idx_msearch_first()` を呼び出す。

5. 10 要素の登録

◇ `int idx_regist (IDX *idxp, void *keyp, LOC *loc)`

指定されたキーと、データ・レコードの位置の要素を登録する。成功の場合0を返す。カーソルは登録した要素を指す。

5. 1 2 削除

◇long idx__del__first__(IDX *idxp, void *key(P)

◇int idx__del__(IDX *idxp, void *keyp, LOC *lo(c)

◇int idx__delete(IDX *idx(P)

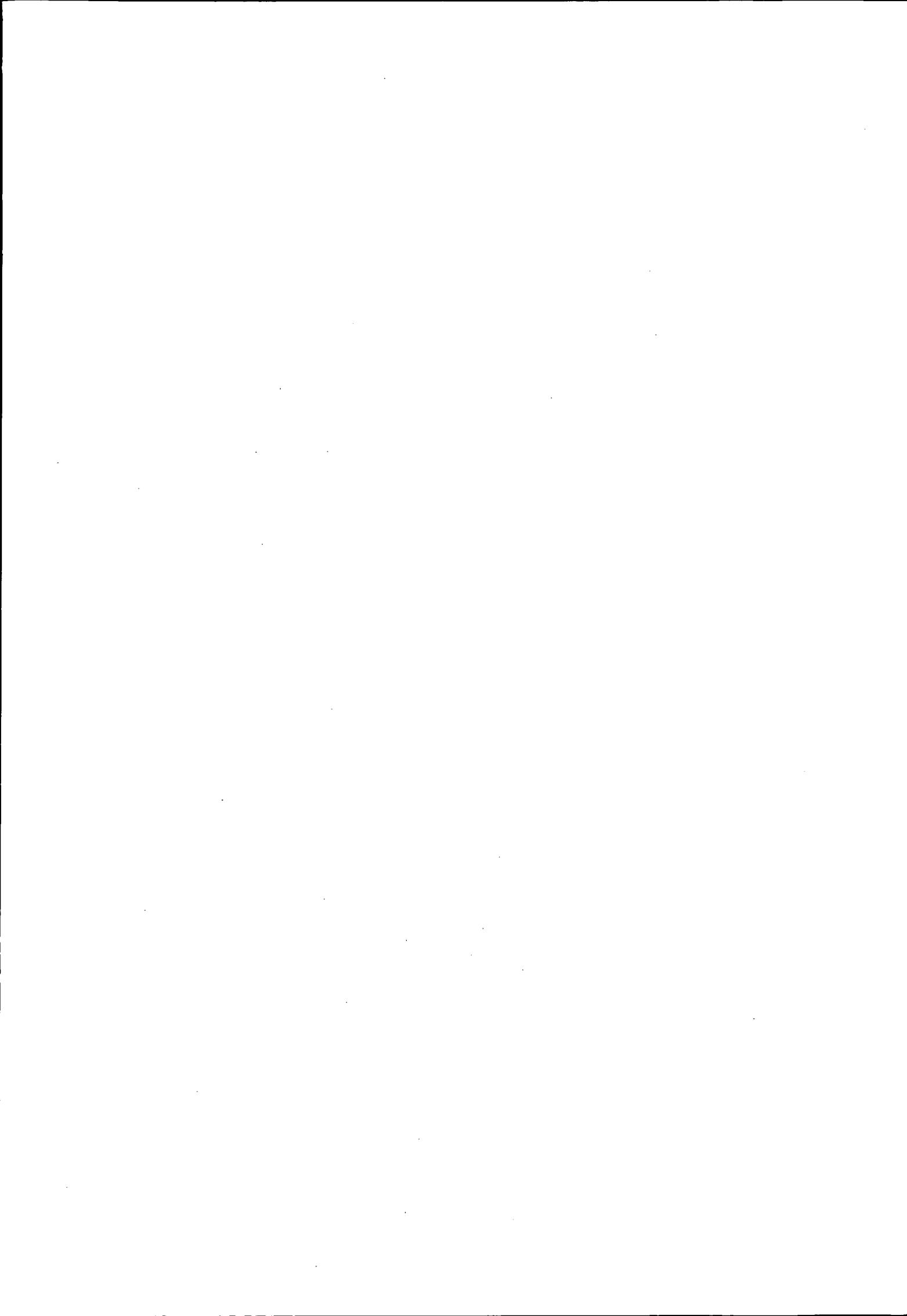
それぞれ、指定キー値をもつ最初の要素を削除する関数、現在カーソルが指している要素を削除する関数。いずれの関数も、カーソルは削除した要素の次の要素を指す。

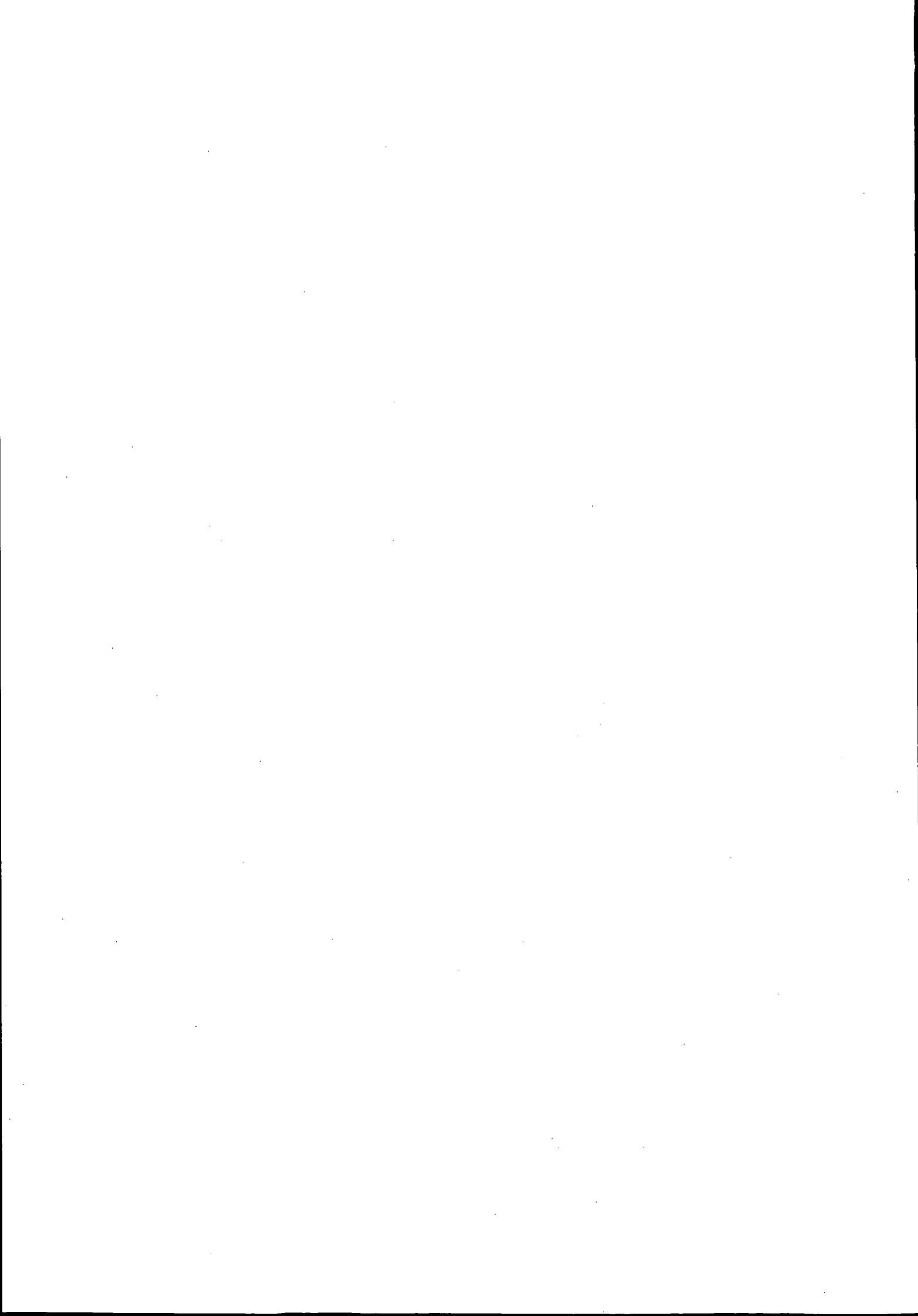
5. 1 3 索引ファイルのダンプ

◇void idx__dump__(IDX *idxp, int *leve(l)

索引ファイルの内容をダンプする関数。

なお、内容表示はC R Tになされる。





—— 禁無断転載 ——

平成5年3月発行

発行 財団法人 データベース振興センター
東京都港区浜松町二丁目4番1号
世界貿易センタービル7階
TEL 03-3459-8581

委託先 (株)日本総合技術研究所
東京都千代田区一番町27ビル4F
TEL 03-3262-8955

印刷所 株式会社 三州社
東京都港区芝大門1-1-21
TEL 03-3433-1481

