

汎用的な属性管理構造に基づく地理情報一元管理手法

3Q-4

佐伯 剛幸

NEC C&C メディア研究所

1 はじめに

近年、OpenGIS(Open Geodata Interoperability Specification)[1]などの標準化により、CORBAやOLE/COMなどの分散コンピューティング環境で定義されたインタフェースに従うことで、GIS(地理情報システム)へさまざまなデータベースシステムを接続することが可能となりつつある。このような背景に基づき、本稿では地理データを構成する属性データと空間データをオブジェクト指向データベース[3]を用いて一元的に管理する分散型地理データベースサーバについて検討する。また、地理データは空間データの構造は共通であるものの属性データは各地理データごとに異なるため、属性データを汎用的に扱える枠組が望まれる。本サーバでは、属性名を文字列で保持して属性データを管理し、スキーマ定義をせずに多様な地理データを格納可能とする。さらに、この枠組みを考慮した属性検索インデックスを提案する。

2 地理データベース

従来、地理データは、名称などの属性データと位置や形状を表す空間データを、前者はリレーショナルデータベース後者はファイルで別々に管理することが一般的であった。近年、オブジェクト指向リレーショナルデータベースやオブジェクト指向データベースを用いて、両者をデータベース内で一元的に管理することが行なわれている。この理由は、別々に管理する場合、図形データを扱うための専用システムを使用して空間データを管理することにより、GISで必要とされる地図関連の処理(地図描画など)を高速に行なえるという利点がある一方で、空間データに対する並行処理制御や問合せができないなどの問題があるためである。このため、本稿ではオブジェクト指向データベースを使用して一元的に管理する方法について検討する。

地理で扱われる属性は、属性名、属性値、属性型を含む。例えば、避難場所という地理データであれば、属性名が面積、属性値が二平方キロメートル、属性型が数値型という属性を持つことが考えられる。属性名の管理方法としては、データベースのスキーマとして管理する方法と文字列データとして管理する方法がある。文字列で

管理する方法は、どのような属性を持つ地理データでもスキーマ定義をせずに管理できるため汎用性が高いが、属性名と属性値からフィーチャー(地物)を検索することになり検索が複雑になる。本稿では、汎用性を重視して属性名を文字列で管理する方法を検討する。さらに、フィーチャーの属性条件による検索を高速化するインデックスを提案する。

3 汎用性の高い地理データベースサーバ

3.1 アーキテクチャ

図1に示すように、本地理データベースサーバは、CORBA経由で空間・属性検索の要求を受け検索結果を返却する。検索機能が、属性条件による検索の場合は属性検索インデックス、空間条件による検索の場合は空間検索インデックスをそれぞれ使用してフィーチャーを検索する。検索結果は転送機能により返却される。

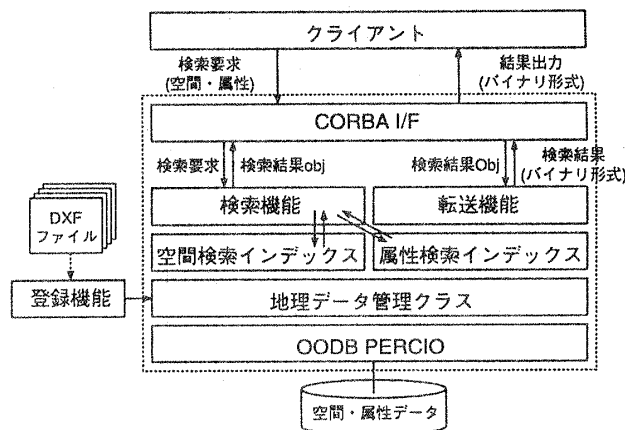


図1: アーキテクチャ

3.2 クラス

本節では地理データ管理クラス(図1参照)の詳細を図2を用いて説明する。図2の各クラスの説明を後に示す。地理データ管理クラスは、属性名を文字列で管理することで汎用性を向上させること(PropertyDefで実現)と、連続するデータを連続領域で管理することによりアクセス速度向上と記憶容量削減をはかること(Property、PropertyDef、Geometryに適用)とを考慮して設計されている。

- ContainerFeatureCollection ... 同一種別のフィーチャーをまとめる、レイヤに相当

- Feature … フィーチャーを表現
- FeatureType … PropertyDefクラスを使用してフィーチャーが持つ属性の構造を表現
- PropertyDef … 属性名と属性型を保持
- Property … フィーチャーが持つ属性の値を保持
- Geometry … フィーチャーの空間データを表現
- Coverage … フィーチャーのカバレッジを表現

ContainerFeatureCollectionはFeatureの集合を管理しており、FeatureTypeがそれらに共通の属性構造を定義する(属性構造についての情報がFeatureTypeに明確に分離される)。FeatureTypeが保持するPropertyDefとFeatureが保持するPropertyはそれぞれ連続領域で管理され、連続領域上の位置により対応つけられる。つまり、ある属性名と同じ位置に存在する属性値がその属性名に対する値となる。PropertyとPropertyDef以外に、Geometryが保持する空間データもポイントの連続領域で管理される。

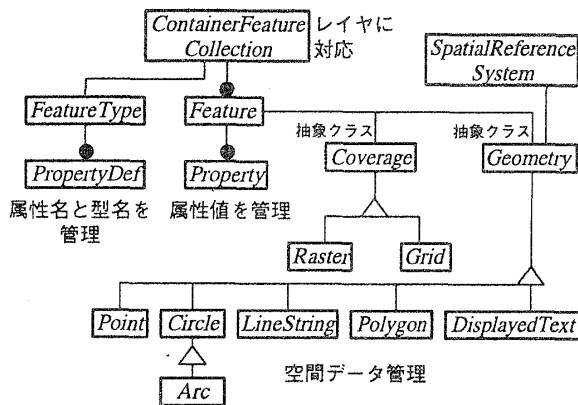


図 2: クラス図

3.3 インデックス

本節では属性検索インデックス(図1参照)の詳細を図3を用いて説明する。通常、属性検索ではレイヤ、属性名、属性値を条件としてフィーチャーを検索する。このため、条件に合致するフィーチャーのみをアクセスできるように絞り込みが高速化のために重要である。

PropertyIndexは、アクセス対象をそれが参照するContainerFeatureCollectionが保持するFeatureに絞り込む(レイヤで絞り込む)。ValueIndexは、さらにアクセス対象を属性名(連続領域上の同じ位置にあるPropertyDefが保持)で絞り込む。上記の絞り込みの結果は属性値とFeatureへの参照を保持するValueIndexElementの集合で管理され、属性値をキーとして二分探索などを行うことで検索条件の属性値に合致するFeatureを高速に取得できる。

属性検索インデックスを使用した検索処理の流れを説明する。レイヤ、属性名、属性値を条件として検索要求が出されると、検索機能(図1参照)は、検索条件のレイヤに対応するContainerFeatureCollectionを探し、それを参照しているPropertyIndexを取得する。さらに、ContainerFeatureCollectionが参照するFeatureTypeをアクセスして検索条件の属性名を保持するPropertyDefの連続領域上の位置を取得し、同じ位置にあるValueIndexをPropertyIndexより取得する。そして、取得したValueIndexから検索条件の属性値に合致するValueIndexElementを探し、それが参照するFeatureを検索結果として返却する。

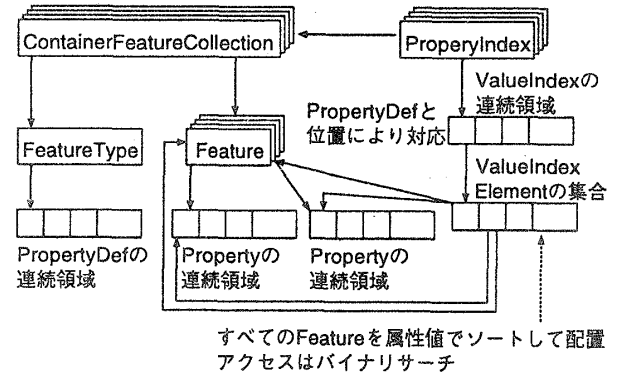


図 3: 属性検索インデックス

4 まとめ

汎用的な属性管理構造を持つ地理情報一元管理手法について述べた。本手法を組み込んだ地理データベースサーバは試作を完了しており、さまざまな地理データをスキーマ定義をせずに問題なく格納できることを確認済みである。また、属性検索に関する簡単な性能測定により、提案したインデックスにより絞り込みが行なわれることを確認している。ただ、属性名をデータとして扱うことでデータ量が増大し、アクセス性能の劣化を引き起こすことも明らかになっている。詳細な評価と上記の問題の解決が今後の課題である。

参考文献

- [1] Open GIS Consortium, Inc., OpenGIS Simple Feature Specification for CORBA. <http://www.opengis.org/>
- [2] N. Adam, A. Gangopadhyay, Database Issues in Geographic Information Systems, Kluwer Academic Publishers, 1997.
- [3] 鶴岡, 他, “オブジェクト指向データベース管理システム PERCIO の開発と今後の課題”, 電子情報通信学会論文誌,D-I, Vol. J79-D-1, No.10, pp.587-596, Oct. 1996.