mapping information has geographic information. Such a commonality is used as a basis of developing an integrated framework that allows a user to access all relevant information on a single platform independent of the data family.

Based on object-oriented technology, the implementation of the multi-data framework of the present invention can be realized by using object pointers to create composite geographic objects with encapsulated data regarding each object type. Topology is required and is specified only for the vector data. However, for a object-oriented framework integrated across data formats, it is necessary to know topological relationships among objects of different data types and coverages to create integrated, composite geographic objects.

The spatial indexing scheme of quad tree indexing and the non-spatial attribute indexing scheme both are used by the present invention. The quad tree indexing scheme is used to find the vicinity of the sought criteria. Then, the other object-oriented indexing schemes based on attribute (nested index, path index, and multi-index) may be used within the selected vicinity of the searching criteria.

Although preferred embodiments of the present invention have been shown and described, it will be appreciated by those skilled in the art that changes may be made in these embodiments without departing from the principle and spirit of the invention, the scope of which is defined in the appended claims and their equivalents.

What is claimed is:

**1**. A method of building and maintaining an object-oriented database from a vector product format (VPF) database, said method comprising:

instantiating objects of the object-oriented database, using the VPF database;

initializing spatial and non-spatial feature data of the object-oriented database;

spatially indexing data among objects across hierarchical levels of the object-oriented database;

updating data of the object-orientated database; and

exporting contents of the updated object-oriented database to the VPF database.

**2**. A method of building and maintaining an object-oriented database from a vector product format (VPF) database, said method comprising:

instantiating objects of the object-oriented database, using the VPF database;

initializing spatial and non-spatial feature data of the object-oriented database;

spatially indexing data among objects across hierarchical levels of the object-oriented database; and

updating spatial and non-spatial data.

**3**. The method according to claim **2**, wherein said initializing spatial and non-spatial feature data creates a feature level having

non-spatial data which provides characteristic properties of each feature, and

spatial data, including primitive data and topological information, which provides spatial relationships between a feature object and other feature objects within a specified coverage; and

wherein related non-spatial and spatial data are directly accessible from the feature object.

**4**. The method according to claim **3**, wherein said updating spatial and non-spatial data includes adding, changing, and deleting feature, primitive, and topological data within the database and further includes updating all object links referencing the feature, primitive, and topological data.

**5**. The method according to claim **2**, wherein said spatially indexing data is applied to one or more databases whose format comprises:

a flat file;

a raster product format;

a vector product format; or

a text format.

**6**. The method according to claim **5**, further comprising retrieving data objects matching a user-specified query based on at least one of the following data characteristics:

feature attributes;

geometrical constraints;

topological constraints; and

geographical constraints.

**7**. The method according to claim **6**, wherein said retrieving data objects includes retrieving:

a flat file;

a raster image;

a VPF feature; or

text data.

* * * * *