COMPONENT-BASED ARCHITECTURE DESIGN FOR FAST INTEROPERABILITY IN SATELLITE IMAGE PROCESSING

CHANG-RAK YOON, KYUNG-OK KIM AND YOUNG-KYU YANG

Satellite Image Information Research Team
Electronics and Telecommunications Research Institute
161 Kajeong-Dong, Yuseong-Gu, Daejeon, 305-350, KOREA
Tel: +82-42-860-5833, Fax: +82-42-860-4844
E-mail: {cryoon, kokim, ykyang}@etri.re.kr

ABSTRACT

Satellite image processing software should be able to support interoperability that is to enable consumers, citizens, large and small businesses and governments to make the fullest possible use of the world's spatial data and spatial processing resources. The goal of interoperability is to perform hands-on research, development, testing, fielding, and validation of potential and existing interoperable geoprocessing technologies. In this paper, component based architecture for fast interoperability is designed to maximize interoperability among the OpenGIS components specified in the OpenGIS abstract specifications and implementation specifications and to mix up the OpenGIS components and the fast access mechanism for high performance.

KEYWORDS : Component, Interoperability, OGC

INTRODUCTION

Recently, many clients need to access, process, analysis spatial information through networks so that interoperability between software vendors and data providers is becoming uprising issue in spatial information services. Especially, development of spatial image processing software has several difficulties to achieve the user's requirement because of very huge data, versatile data formats and different architectures between software vendors. Nevertheless, as computing environment is shifting to distributed networks, the developers want to establish a standard manners to support interoperability.

The component based development methodology has been concentrated in software engineering as new paradigm to produce, select, evaluate and integrate software. The component based development methodology has two advantages, one is reusability of pre-implemented executable components and the other is location transparency in distributed networking environment. The component software which independently runs with other components, can be easily accessed by any one who knows the defined interfaces of the components. The developers efficiently provide the interoperability on distributed networking environment and the reusability of pre-developed components.

So, OpenGIS consortium has announced the standards to support interoperability with component based development methodology. In this paper, we introduce OpenGIS Grid Coverages Implementation Specification and provide the means for unprecedented fast interoperability between systems that produce and use satellite images of the Earth, digital aerial photos, digital elevation data and other kinds of data represented in a grid cell or "raster" image coordinate system that is tiled or not tiled to an Earth coordinate system. Also, component based architecture for fast interoperability is designed to establish integrated software.

RELATED STUDIES

In 1994, the OpenGIS consortium was founded in response to wide-spread recognition of the problem of non-interoperability.[1-2] OpenGIS grid coverage implementation specification is released through the OpenGIS standards development process. This specification provides interfaces for basic inter-system image access and for basic kinds of analysis, such as histogram calculation, image covariance and other statistical measurements. That is, the grid coverage specification enables GIS systems with grid coverage components from heterogeneous vendors of image and raster software to query one another over a network to copy data or operate on data. It does not matter that the systems in the client and server roles use different native data formats or different internal command structures. The open interface they share enables them to communicate.

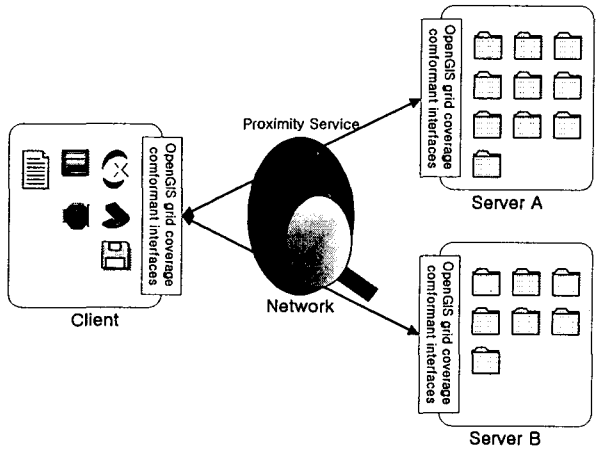

Figure 1. Interoperability using OpenGIS grid coverage specification conformant interfaces

The grid coverage specification embraces various file formats, various color models, variable value sequencing, pixel ordering and all standard grid geometries and spatial referencing systems consistent with geometries and spatial referencing in the OpenGIS simple feature specification. It is consistent with geometries and spatial referencing in the OpenGIS simple feature specification, which addresses vector data

The architecture of the grid coverage is composed of three component packages as shown in Figure 2. The specification includes a package for the general coverage specification (CV), a package specifically for grid coverages (GC) and a package for grid coverage processing (GP). The GP package is optional and not required for an OpenGIS compliant implementation.

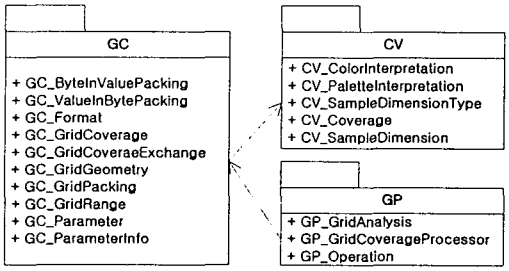

Figure 2. OpenGIS grid coverage component packages

The grid coverage specification can accommodate grid coverages with the following characteristics: variable number of bits per grid value, multi-bands, multi-dimensions, various band ordering, variable number of no data values, various color models, reduced resolution data sets, grid coordinate systems, tiling data sets and various color palettes. To help interoperability, many interfaces contained in this architecture have been made immutable. The assumption is that the objects supporting these interfaces should be immutable. This means that the objects should not change after a factory has created them. In a GC_GridCoverage interface, only the cell values may change. The size, geometry, interpolation, sequencing etc should never change. However, clients should be aware that the grid coverage object behind the GC_GridCoverage interface may be adapted from other GC_GridCoverage interfaces, so changing the grid point values through one GC_GridCoverage could change the grid point values in others. To allow client applications to anticipate adapted grid coverage being changed indirectly, the source grid geometry object for adapted grid geometry objects can be traced.

OpenGIS Coverage Package

The CV_Coverage interface is abstract interface providing access to an OpenGIS coverage. The essential property of coverage is to be able to generate a value for any point within its domain however coverage is represented internally. A coverage may be represented by a set of polygons which exhaustively tile a plane, a grid of values, a mathematical function or combination of these. Figure 3 illustrates the coverage package.

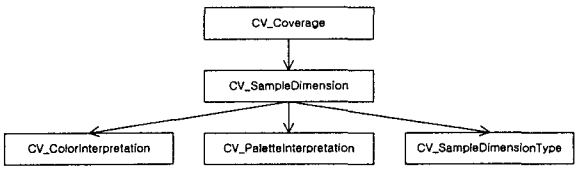

Figure 3. Coverage package

OpenGIS Grid Coverage Package

The grid coverage interface would represent the basic implementation which provides access to grid coverage data. A GC_GridCoverage interface provides the ability to update grid values. A basic read-only implementation would be fairly easy to implement. The grid coverage package is illustrated in figure 4.

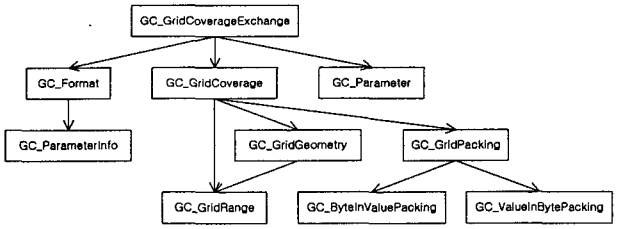

Figure 4. Grid coverage package

OpenGIS Grid Coverage Processor Package

The grid coverage processor package is composed of optional interfaces and provides operations for different ways of accessing the grid coverage values as well as image processing functionality. The list of available processing operations is implementation dependent. The interface has a discovery mechanism to determine the available processing operations. This grid coverage processor package is shown in Figure 5.

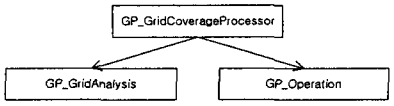

Figure 5. Grid coverage processor package

COMPONENT-BASED ARCHITECTURE DESIGN FOR FAST GRID PROCESSING

The OpenGIS consortium provides the three component packages which can access and process grid data. In these component packages, the grid coverage processor package may mainly contribute development for fast grid processing application. The processing operations in GP_GridCoverageProcessor interface will transform values within a specified sample dimensions and leave the values in other sample dimensions unaffected. The modified sample dimensions may also change its type but the actual underlying grid data remains unchanged.

The grid coverage processor interfaces have been designed to allow the adaptations to be done in a pipe-lined manner. The interfaces operate on GC_GridCoverage to create new a GC_GridCoverage. The interface does not need to make a copy of the source grid data. Instead, it can return a grid coverage object which applies the adaptations on the original grid coverage whenever a block of data is requested. In this way, a pipeline of several grid coverages can be constructed cheaply.

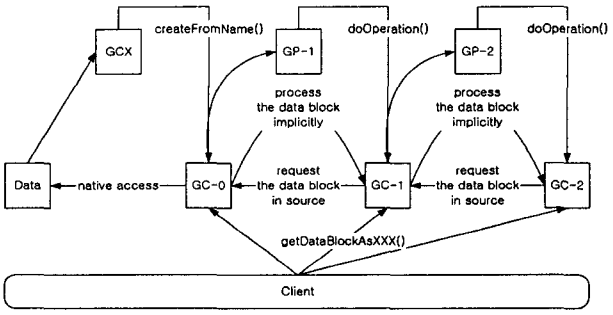

Figure 6. Pipe-lined Grid Coverage Processing

In the pipe-lined processing mechanism, grid coverage processors implemented by various developers should provide GP_GridCoverageProcessor interface which has not copies of grid data. In figure 6, client can create grid coverage object by calling createFromName() in GC_GridCoverageExchange interface and then new grid coverage object by calling doOperation() in GP_GridCoverageProcessor interface. In this way, whenever client requires grid data, grid coverage object requests the data block in source grid coverage object and then the requested source grid coverage object processes its implicit operation to return the processed data through pipeline. Especially, if a grid coverage processor should execute operations with statistical values such as minimum value or maximum value, it is recommended that client creates GP_GridAnalysis object inherited from GC_GridCoverage interface. The GP_GridAnalysis interface can efficiently process statistical operations such as linear enhancement and histogram equalization because various statistical values can be implicitly included in the GC_GridAnalysis interface. Figure 7 illustrates the creation a grid coverage from a grid coverage exchange and invoking a processing operation after discovering the available operations. Grid values are retrieved from the modified grid coverage and the grid coverage processor fetches grid values from the source grid coverage as required.

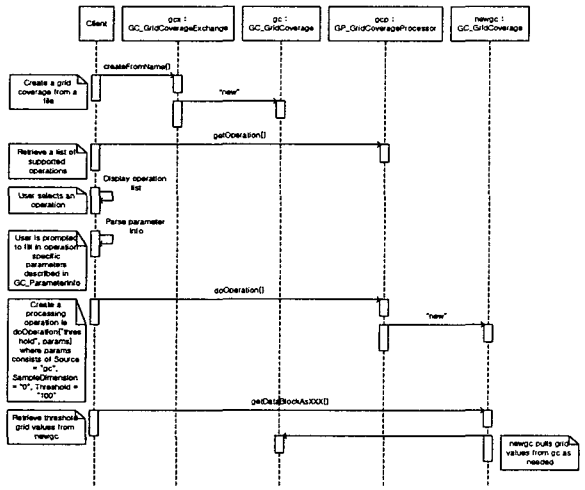

Figure 7. Grid coverage processing using the discovery mechanism

CONCLUSIONS

OpenGIS grid coverage is composed of three component packages which access grid data to support interoperability between data providers and software vendors. Each grid coverage package includes various interfaces to do specific functionalities and relates its interfaces according to its purpose. OpenGIS consortium suggested the recommended definition of grid coverage component packages but not its detailed implementation so that developers for OpenGIS grid coverage conformant interfaces should innovate details of properties and methods for grid coverage interfaces. In this paper, we designs the pipe-lined grid coverage processing that implements the conceptual definitions of OpenGIS grid coverage component packages and suggests the fundamental architecture to follow the OpenGIS grid coverage implementation specification.

REFERENCES

[1] OpenGIS Consortium, Inc., "OpenGIS Implementation Specification : Grid Coverage", Rev. 1.0, 2001

[2] OpenGIS Consortium, Inc., The OpenGIS Guide, 3rd Ed., 1998